Advances in Intelligent Systems and Computing 664

S. S. Agrawal
Amita Dev
Ritika Wason
Poonam Bansal *Editors*

Speech and Language Processing for Human-Machine Communications

Proceedings of CSI 2015

Springer

Advances in Intelligent Systems and Computing

Volume 664

Series editor

Janusz Kacprzyk, Polish Academy of Sciences, Warsaw, Poland
e-mail: kacprzyk@ibspan.waw.pl

The series "Advances in Intelligent Systems and Computing" contains publications on theory, applications, and design methods of Intelligent Systems and Intelligent Computing. Virtually all disciplines such as engineering, natural sciences, computer and information science, ICT, economics, business, e-commerce, environment, healthcare, life science are covered. The list of topics spans all the areas of modern intelligent systems and computing.

The publications within "Advances in Intelligent Systems and Computing" are primarily textbooks and proceedings of important conferences, symposia and congresses. They cover significant recent developments in the field, both of a foundational and applicable character. An important characteristic feature of the series is the short publication time and world-wide distribution. This permits a rapid and broad dissemination of research results.

More information about this series at http://www.springer.com/series/11156

S. S. Agrawal · Amita Dev
Ritika Wason · Poonam Bansal
Editors

Speech and Language Processing for Human-Machine Communications

Proceedings of CSI 2015

Springer

Editors

S. S. Agrawal
KIIT
Gurgaon, Haryana
India

Amita Dev
Bhai Parmanand Institute of Business Studies
New Delhi, Delhi
India

Ritika Wason
MCA Department
Bhrati Vidyapeeth's Institute of Computer Applications and Management (BVICAM)
New Delhi, Delhi
India

Poonam Bansal
Maharaja Surajmal Institute of Technology
GGSIP University
New Delhi, Delhi
India

ISSN 2194-5357 ISSN 2194-5365 (electronic)
Advances in Intelligent Systems and Computing
ISBN 978-981-10-6625-2 ISBN 978-981-10-6626-9 (eBook)
https://doi.org/10.1007/978-981-10-6626-9

Library of Congress Control Number: 2017956742

The use of general descriptive names, registered names, trademarks, service marks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.
The publisher, the authors and the editors are safe to assume that the advice and information in this book are believed to be true and accurate at the date of publication. Neither the publisher nor the authors or the editors give a warranty, express or implied, with respect to the material contained herein or for any errors or omissions that may have been made. The publisher remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Printed on acid-free paper

This Springer imprint is published by Springer Nature
The registered company is Springer Nature Singapore Pte Ltd.
The registered company address is: 152 Beach Road, #21-01/04 Gateway East, Singapore 189721, Singapore

Preface

The last decade has witnessed remarkable changes in IT industry, virtually in all domains. The 50th Annual Convention, CSI-2015, on the theme "Digital Life" was organized as a part of CSI@50, by CSI at Delhi, the national capital of the country, during December 2–5, 2015. Its concept was formed with an objective to keep ICT community abreast of emerging paradigms in the areas of computing technologies and more importantly looking at its impact on the society.

Information and Communication Technology (ICT) comprises of three main components: infrastructure, services, and product. These components include the Internet, infrastructure-based/infrastructure-less wireless networks, mobile terminals, and other communication mediums. ICT is gaining popularity due to rapid growth in communication capabilities for real-time-based applications. "Nature Inspired Computing" is aimed at highlighting practical aspects of computational intelligence including robotics support for artificial immune systems. CSI-2015 attracted over 1500 papers from researchers and practitioners from academia, industry, and government agencies, from all over the world, thereby making the job of the Programme Committee extremely difficult. After a series of tough review exercises by a team of over 700 experts, 565 papers were accepted for presentation in CSI-2015 during the 3 days of the convention under ten parallel tracks. The Programme Committee, in consultation with Springer, the world's largest publisher of scientific documents, decided to publish the proceedings of the presented papers, after the convention, in ten topical volumes, under ASIC series of Springer, as detailed hereunder:

1. Volume # 1: ICT based Innovations
2. Volume # 2: Next Generation Networks
3. Volume # 3: Nature Inspired Computing
4. Volume # 4: Speech and Language Processing for Human-Machine Communications
5. Volume # 5: Sensors and Image Processing
6. Volume # 6: Big Data Analytics

7. Volume # 7: Systems and Architecture
8. Volume # 8: Cyber Security
9. Volume # 9: Software Engineering
10. Volume # 10: Silicon Photonics & High Performance Computing

We are pleased to present before you the proceedings of Volume # 4 on “Speech and Language Processing for Human-Machine Communications.” The idea of empowering computers with the power to understand and process human language is a pioneering research initiative. The main goal of SLP field is to enable computing machines to perform useful tasks through human language like enabling and improving human–machine communication. The past two decades have witnessed an increasing development and improvement of tools and techniques available for human–machine communication. Further, a noticeable growth has also been witnessed in the tools and implementations available for natural language and speech processing.

In today’s scenario, developing countries have made a remarkable progress in communication by incorporating the latest technologies. Their main emphasis is not only on finding the emerging paradigms of information and communication technologies but also on its overall impact on the society. It is imperative to understand the underlying principles, technologies, and ongoing research to ensure better preparedness for responding to upcoming technological trends. Keeping the above points in mind, this volume is published, which would be beneficial for researchers of this domain.

The volume includes scientific, original, and high-quality papers presenting novel research, ideas, and explorations of new vistas in speech and language processing such as speech recognition, text recognition, embedded platform for information retrieval, segmentation, filtering and classification of data, and emotion recognition. The aim of this volume is to provide a stimulating forum for sharing knowledge and results in model, methodology, and implementations of speech and language processing tools. Its authors are researchers and experts in these domains. This volume is designed to bring together researchers and practitioners from academia and industry to focus on extending the understanding and establishing new collaborations in these areas. It is the outcome of the hard work of the editorial team, who have relentlessly worked with the authors and steered them up to compile this volume. It will be a useful source of reference for the future researchers in this domain. Under the CSI-2015 umbrella, we received over 100 papers for this volume, out of which 23 papers are being published, after rigorous review processes carried out in multiple cycles.

On behalf of the organizing team, it is a matter of great pleasure that CSI-2015 has received an overwhelming response from various professionals from across the country. The organizers of CSI-2015 are thankful to the members of the *Advisory Committee, Programme Committee, and Organizing Committee* for their all-round guidance, encouragement, and continuous support. We express our sincere gratitude to the learned *Keynote Speakers* for their support and help extended to make this event a grand success. Our sincere thanks are also due to our *Review Committee*

Members and the *Editorial Board* for their untiring efforts in reviewing the manuscripts and giving suggestions and valuable inputs in shaping this volume. We hope that all the participants/delegates will be benefitted academically and wish them all the best for their future endeavors.

We also take the opportunity to thank the entire team from Springer, who have worked tirelessly and made the publication of the volume a reality. Last but not least, we thank the team from Bharati Vidyapeeth's Institute of Computer Applications and Management (BVICAM), New Delhi, for their untiring support, without which the compilation of this huge volume would not have been possible.

Gurgaon, India	S. S. Agrawal
New Delhi, India	Amita Dev
New Delhi, India	Ritika Wason
New Delhi, India	Poonam Bansal
March 2017	

The Organization of CSI-2015

Chief Patron

Padmashree Dr. R. Chidambaram, Principal Scientific Advisor, Government of India

Patrons

Prof. S. V. Raghavan, Department of Computer Science, IIT Madras, Chennai
Prof. Ashutosh Sharma, Secretary, Department of Science and Technology, Ministry of Science of Technology, Government of India

Chair, Programme Committee
Prof. K. K. Aggarwal, Founder Vice Chancellor, GGSIP University, New Delhi

Secretary, Programme Committee
Prof. M. N. Hoda, Director, Bharati Vidyapeeth's Institute of Computer Applications and Management (BVICAM), New Delhi

Advisory Committee

Padma Bhushan Dr. F. C. Kohli, Co-Founder, TCS
Mr. Ravindra Nath, CMD, National Small Industries Corporation, New Delhi
Dr. Omkar Rai, Director General, Software Technological Parks of India (STPI), New Delhi
Adv. Pavan Duggal, Noted Cyber Law Advocate, Supreme Courts of India
Prof. Bipin Mehta, President, CSI

Prof. Anirban Basu, Vice President-cum-President Elect, CSI
Shri Sanjay Mohapatra, Secretary, CSI
Prof. Yogesh Singh, Vice Chancellor, Delhi Technological University, Delhi
Prof. S. K. Gupta, Department of Computer Science and Engineering, IIT Delhi, Delhi
Prof. P. B. Sharma, Founder Vice Chancellor, Delhi Technological University, Delhi
Mr. Prakash Kumar, IAS, Chief Executive Officer, Goods and Services Tax Network (GSTN)
Mr. R. S. Mani, Group Head, National Knowledge Networks (NKN), NIC, Government of India, New Delhi

Editorial Board

Contents

About the Editors

Dr. S. S. Agrawal is a world-renowned scientist and a teacher in the area of Acoustic Speech and Communication. He obtained his Ph.D. degree in 1970 from the Aligarh Muslim University, India. He has a research experience of about 45 years at the Central Electronics Engineering Research Institute (CEERI), Pilani, and subsequently worked as Emeritus Scientist at the Council of Scientific and Industrial Research (CSIR) and as Advisor at the Centre for Development of Advanced Computing (CDAC), Noida. He was a Guest Researcher at the Massachusetts Institute of Technology (MIT), Ohio State University, and University of California, Los Angeles (UCLA), USA. His major areas of interest are Spoken Language Processing and Development of Speech Databases, and he has steered many national and international projects. He has published a large number of papers, guided many Ph.D. students, and received honors and awards in India and abroad. He is currently working as Director General at KIIT Group of Colleges, Gurgaon, Haryana.

Dr. (Mrs.) Amita Dev obtained her B.Tech. degree from Panjab University, Chandigarh, and completed her postgraduation from the Birla Institute of Technology and Science (BITS), Pilani, India. She obtained her Ph.D. degree from the Delhi College of Engineering under University of Delhi in the area of Computer Science. She is a Fellow of the Institution of Electronics and Telecommunication Engineers (IETE) and a Life Member of the Indian Society for Technical Education (ISTE) and Computer Society of India (CSI). She has more than 30 years of experience and is presently working as the a Principal at Ambedkar Institute of Technology, Delhi and Bhai Parmanand Institute of Business Studies, Delhi, under the Department of Training and Technical Education, Government of National Capital Territory (NCT) of Delhi. She has been awarded the "National Level Best Engineering Teachers Award" in the year 2001 by ISTE for her significant contribution in the field of Engineering and Technology. She has also been awarded the "State Level Best Teacher Award" by the Department of Training and Technical Education, Government of NCT of Delhi. She is a recipient of the "National Level Young Teachers Award" for pursuing advance research in the field of Speech

Recognition. She has published more than 45 papers in leading national and international Journals and in conference proceedings of leading conferences. She has written several books in the area of Computer Science and Engineering.

Dr. Ritika Wason has completed her Ph.D. degree in Computer Science from the Sharda University, Delhi, and obtained her postgraduation from Indraprashtha University (IPU, now known as Guru Gobind Singh Indraprastha University). She is a Life Member of the Indian Society for Technical Education (ISTE) and Computer Society of India (CSI). She has almost 10 years of teaching experience and is presently working as Assistant Professor, at Bharati Vidyapeeth's Institute of Computer Applications and Management (BVICAM), New Delhi. She has published more than 20 papers in leading national and international journals and in conference proceedings of leading conferences. She has also authored several books in the area of Computer Science and Engineering.

Prof. Poonam Bansal is Acting Director at the Maharaja Surajmal Institute of Technology (MSIT), a prestigious institute affiliated to the Guru Gobind Singh Indraprastha University (GGSIPU), New Delhi. She has 24 years of wide and rich experience in industry, teaching, and research. She received her B.Tech. and M. Tech. degrees from the Delhi College of Engineering, Delhi, and obtained Ph.D. degree from GGSIPU, New Delhi. She has published more than 25 research papers in peer-reviewed journals and conferences of national and international repute. Her areas of interest include Speech Technology, Soft Computing, and Computer Networking.

AC: An Audio Classifier to Classify Violent Extensive Audios

Anuradha Pillai and Prachi Kaushik

Abstract This paper presents audio-based classifier to classify the audio into four audio classes like music, speech, gunshots and screams. The audio signals are divided into frames, and various frame-level time and frequency features are calculated for the segment of audio. The classification rules are based on the combination of statistics value calculated for each feature. The classifier takes an unknown segment of audio, applies the classification rules and outputs the label for particular audio. The audio classifier performs with an effective recall rate of 84%.

Keywords Audio classifier · Coefficient of variation · Analyser · Extract features

1 Introduction

The growth of the multimedia data which is accessible though the World Wide Web (WWW), so there is a need for content-based retrieval of information indexing of the audio visual data. There have been several methods for the classification of the audio visual or images into a particular predefined class. Audio classification is an important area of research which has focused on classification of music genres, recognition of the musical instruments which are played in the audio, speaker identification by means of the audio signals, recognition of the emotion from the speech data or the musical audio. The audio data is rich and informative source of extraction of the type of content which involves content-based classification of the audio signals. Audio signals are classified into predefined classes such as violent and non-violent content. After analysis of several violent audio data, it was found that such videos contained continuous voices of gunshots, explosions and human

A. Pillai (✉)
CE Department, YMCAUST, Faridabad, India
e-mail: anuangra@yahoo.com

P. Kaushik
BCA Department, DAV Centenary College, Faridabad, India
e-mail: prachikaushik.4@gmail.com

S. S. Agrawal et al. (eds.), *Speech and Language Processing for Human-Machine Communications*, Advances in Intelligent Systems and Computing 664,
https://doi.org/10.1007/978-981-10-6626-9_1

screaming [1–3]. Violence in the audio data can also be detected by the use of several hate and abusive words due to anger. This violence is called as oral violence which is conveyed by using certain words to show anger.

In this research, several audio features such as time-domain and frequency-domain features are used to classify the audio segment into particular predefined categories [1, 4]. The statistics chosen for this work is coefficient of variation (CV) which proves to be effective in audio classification. A new feature is used to distinguish and assign music and speech labels to the audio signal that is the percentage of the silence intervals (SI). It has been observed that speech has a higher SI value because speaker pauses while speaking sentences, but music is a tonal. The recall rate of the audio-based classifier is approximately 84%.

2 Related Work and Research Contributions

The following literature survey is done mainly on the topic of audio classification of violent sounds using a set of features extracted from audio file.

1. Vozarikova et al. [4] present a methodology to detect dual gunshots in noisy environment using features such as MFCC, MELSPEC, skewness, kurtosis and ZCR. The combination of different features was evaluated by the HMM classification technique.
2. Pikrakis [3] identified gunshots by dynamic programming and Bayesian network. The posterior probabilities were calculated by combing the decisions from a set of Bayesian network combiners, and 80% of the gunshots were correctly detected.
3. Gerosa et al. [5] in their approach trained two parallel GMM classifiers to differentiate gunshots and screams from noisy environment. A set of 47 audio features were used for the classification, and the proposed system guarantees a precision of 90%.
4. Giannakopoulos et al. [1] proposed a methodology to detect violent scenes in movies using twelve audio features and visual features combined together. The video features included certain motion specific features such as average motion, motion oriented variance and detection features for the face detection in the scenes. The performance of the system is 83%, and only 17% of the scenes are not detected.
5. Giannakopoulos, Kosmopoulous [6] used time-domain and frequency-domain features along with the SVM classifier to detect violence content. The recall rate was 90.5% which could be further improved by MFCC coefficients.
6. Zou et al. [2] in this paper propose a text-, audio-, visual-based classification. The first stage is a text-based classifier to identify potential movie segments. The second stage used a combination of audio and visual cues to detect violence.

Table 1 gives brief information related to the various text, audio, visual features extracted in respective research papers. It also highlights the classification approaches used in the classification of audio content.

Table 1 Research contributions in the area of audio classification

Research paper	Features			Classification approach
	Text	Audio	Visual	
Uzkent et al.	×	New set of pitch features and auto-correlation coefficients	×	SVM Radial basis function neural n/w SVM with Gaussian kernel (best performance)
Vozarikova et al.	×	MFCC, MELSPEC, skewness, kurtosis, ZCR	×	HMM classifier
Pikrakis	×	Entropy, ZCR, 3 MFCC, roll-off, pitch	×	Bayesian n/w classifier + dynamic programming
Gerosa et al.	×	47 features: ZCR, 4 spectral moments, 30 MFCC, slope of spectrum, decrease, roll-off, periodicity, correlation slope	×	Two parallel GMM classifier
Giannakopoulos et al.	×	12 features: ZCR, entropy, 3 MFCC, roll-off, zero pitch ratio, chroma features, spectrogram features	Motion features: Average motion Motion oriented variance Detection features: Face detection	
Kosmopoulos	×	Entropy, ZCR, signal amplitude, energy, spectral flux, spectral roll-off	×	SVM
Zou et al.	Yes	Energy entropy	Motion intensity Colour of flame Colour of blood Length of shot	SVM

3 Audio Classification

This module of the proposed work inputs a segment of the audio and divides it into frames of 100 ms. For each frame, time-domain features and frequency-domain features are extracted for the classification of audio into four classes music, speech, gunshots, scream. Figure 1 shows the architecture for the audio classification of the audio segments into four classes. The next section discusses the working of each component and the classification rules which are used to classify an audio segment correctly.

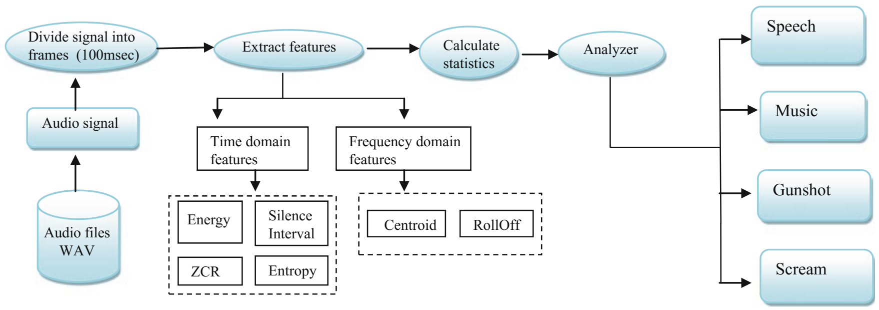


Fig. 1 Architecture for audio-based classifier

3.1 Repository of Audio Files

Audio is a sound which the normal human ear can listen. The audible frequency range for the human ear is between 20 and 20,000 Hz. The audio files are in wav format with a sampling rate of 44.1 kHz. Sampling rate is the number of samples the audio carries in 1 s, which is measured in Hz or kHz.

3.2 Audio Signal

The audio signal for each segment is plotted in MATLAB, and the graphical representation of each is shown below in Fig. 2. The segments below have a unique pattern which can be distinguished easily by human eye, but various features need to be extracted for the computer to give the correct class for the audio.

3.3 Divide Signal into Frames

The signal is broken down into smaller frames of 100 ms. The frame time is multiplied by the sampling rate fs to calculate length of frame.

$$\boxed{\text{Number of frames} = \text{Length of Audio Signal}/\text{Length of one frame}}$$

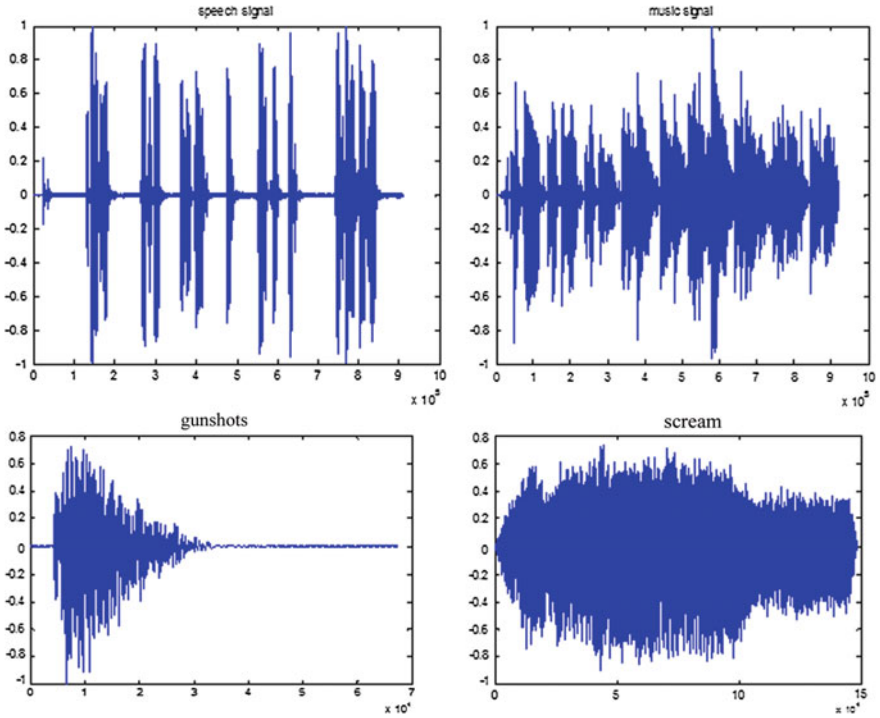


Fig. 2 Plot of audio signals for speech, music, gunshots, scream

3.4 Extract Features

The audio classification is based on finding patterns which are identified by the set of features used in this work. Hence, feature extraction plays a central role for audio analysis and classification into certain classes. The process involves computation of numerical values and representations which can characterize an audio signal.

Time-Domain Audio Features Time-domain features analyse the signal with respect to the time frame. It gives an overview of the signal changes over time domain. The features which are extracted directly from time represent the energy changes in the signal. Therefore, they can be used for audio signal identification. These audio features are simple in nature.

Energy Let $x_i(n)_{n=1}^{N}$ be the ith frame of length N containing audio samples from 1 to N. Then, for each frame i the energy is calculated according to (1):

$$E(i) = \frac{1}{N}\sum_{n=1}^{N}|x_i(n)|^2 \qquad (1)$$

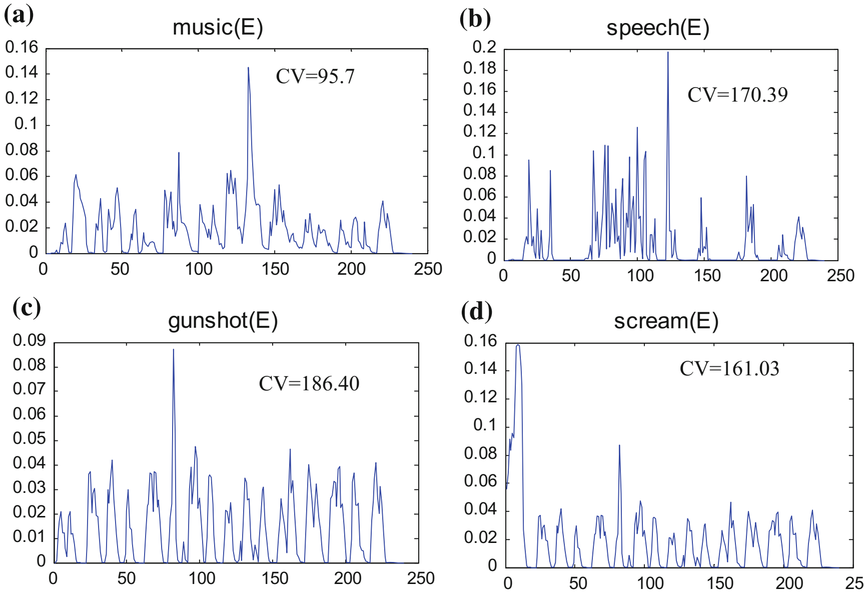


Fig. 3 Energy waveforms (*E*) for **a** music, **b** speech, **c** gunshot, **d** scream

The variation of energy (CV) in the speech segment is higher than music signal as its energy alternates from high to low. The statistics calculated for energy is the CV (coefficient of variation). Energy waveform (*E*) of (a) music (b) speech (c) gunshot (d) scream is shown in Fig. 3. According to the CV values, the order of audio signal is music < scream < speech < gunshot. Gunshot has the highest value for CV and music the lowest CV.

Zero-Crossing Rate It is abbreviated as ZCR and measures the number of times the signal alternates from positive to negative and back to positive. The ZCR value of periodic signal is less as compared to noisy signal. The formula (2) to calculate ZCR is given below:

$$Z(i) = \frac{1}{2N}\sum_{n=1}^{N}\left|\operatorname{sgn}[x_i(n)] - \operatorname{sgn}[x_i(n-1)]\right| \tag{2}$$

The CV value of the ZCR sequence of the speech segment is higher than the music segment due to abrupt changes from positive to negative. Statistics calculated for ZCR is CV and mean.

Figure 4 depicts the waveform for the ZCR values for music, speech, gunshot, scream. According to the experimentation value of ZCR_{CV} is in the following order: scream < music < speech < gunshot. The highest value of ZCR_{CV} is for gunshot and the lowest is for scream. If we arrange the series in increasing order of mean values, the order is: music < speech < scream < gunshot.

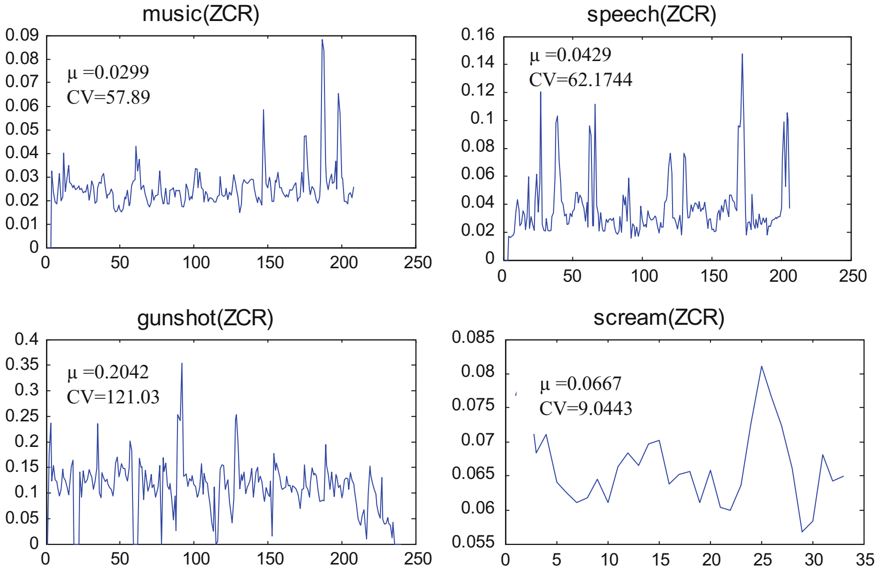


Fig. 4 Waveform for the ZCR values for music, speech, gunshot, screams

Energy Entropy Energy entropy is a time-domain feature that takes into account abrupt changes in the energy level of an audio signal. Each frame is divided into K fixed duration sub-frames. The normalized energy e_j^2 is calculated (3) for each sub-frame by dividing the sub-frame's energy, by the whole frame's energy.

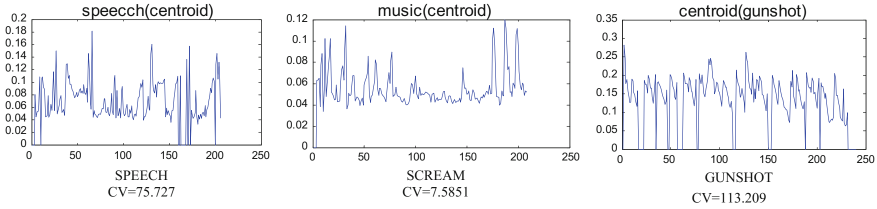
$$e_j^2 = \frac{\text{Esub-frame}_j}{\text{Eshort frame}_i} \tag{3}$$

The En(i) energy entropy of frame$_i$ is calculated below (4)

$$\text{En}(i) = -\sum_{i=0}^{K} e_j^2 \cdot \log_2(e_j^2) \tag{4}$$

The statistics value of the energy entropy is taken as the coefficient of variation. According to the experimentation, the audio signals with abrupt changes have a higher value for CV. Gunshots and speech have larger value for the coefficient of variation compared to screams and music.

Frequency-Domain Audio Features Frequency-based features along with the time-domain feature make an effective combination for the task of classification of audio in different classes. This domain refers to the analysis of the audio signal based on the frequency values. This domain analysis gives information regarding



SPEECH
CV=75.727

SCREAM
CV=7.5851

GUNSHOT
CV=113.209

Fig. 5 Spectral centroid for speech, scream and gunshot

the signal's energy distribution over a range of frequencies. The Fourier transform is a mathematical operation which converts a time-domain signal into its corresponding frequency domain.

Spectral Centroid It is a measure used in digital signal processing to identify a spectrum. It signifies the concentration of the centre of mass of the spectrum. Spectral centroid for screams has a low deviation, and speech signals have highly variated spectral centroid. Figure 5 shows that gunshot has the highest CV value and scream has the lowest CV value; hence, the order is: scream < speech < gunshot.

Eq. (5) is given below:

$$C_i = \frac{\sum_{k=1}^{N}(K+1)X_i(k)}{\sum_{k=1}^{N}X_i(k)} \tag{5}$$

Spectral Roll-Off The frequency below which 90% of the magnitude distribution of the spectrum is concentrated is called spectral roll-off. This feature is described as follows: if the mth DFT coefficient corresponds to the spectral roll-off of the ith frame, then the following equation holds:

$$\sum_{k=1}^{m} X_i(k) = C\sum_{k=1}^{N} X_i(k) \tag{6}$$

where C is the adopted percentage. It has to be noted that the spectral roll-off frequency is normalized by N, in order to achieve values between 0 and 1. Spectral roll-off measures the spectral shape of an audio signal, and it can be used for discriminating between voiced and unvoiced speech. It can be seen that this statistics is lower for the music part, while this value is higher for environmental sounds. The calculated mean and median of the spectral roll-off is shown in Table 2.

Table 2 Mean and median value corresponding to roll-off

Audio	Mean	Median
Music	0.32	0.29
Speech	0.38	0.37
Gunshot	0.68	0.72
Scream	0.35	0.35

3.5 Calculate Statistics

The calculate statistics phase is an important part in the classification procedure. Values of statistics are formulated in form of rules. Coefficient of variation has been used as a major statistics in the proposed work shown in Table 2.

$$*\mathrm{CV} = \text{Coefficient of Variation} = (\text{Standard Deviation}/\text{Mean}) * 100$$

If CV (A) > CV (B), there are some points to note:

1. B is more consistent.
2. B is more stable.
3. B is more uniform.
4. A is more variable.

The CV values of every feature are able to distinguish among the predicted classes and hence useful for the classification work.

3.6 Analyser

The new audio is to be assigned a label from the following labels {music, speech, scream, gunshots}. Before the function of the analyser, the statistics (refer Table 3) used for each feature is calculated.

1. IF $E_{\mathrm{CV}}(a) > 100$ && ($\mathrm{ZCR}_{\mathrm{CV}}(a) > 100$ $\|$ $\mathrm{ZCR}_{\mathrm{Mean}} > 0.1000$) && $\mathrm{En}_{\mathrm{CV}}(a) > 200$ && $C_{\mathrm{CV}}(a) > 100$ $\|$ ($\mathrm{RO}_{\mu}(a) > 0.50$ $\|$ $\mathrm{RO}_{\mathrm{med}}(a) > 0.50$) → GUNSHOT,

Table 3 Statistics

Feature	Statistics
Energy	CV
ZCR	CV, Mean (μ)
Entropy	CV
Centroid	CV
Roll off	Mean, median

2. IF ($E_{\mathrm{CV}}(a) > 100$ && ($\mathrm{ZCR}_{\mathrm{CV}}(a) < 100$ || $\mathrm{C}_{\mathrm{CV}} < 100$)), entropy of the audio is calculated.
   If $\mathrm{entropy}_{\mathrm{CV}} > 200$ && $\mathrm{ZCR}_{\mathrm{Mean}} > 0.1000$ → GUNSHOT audio with multiple shots,
3. IF $E_{\mathrm{CV}} > 100$ && $\mathrm{ZCR}_{\mathrm{CV}} < 100$ audio may belong to any of the three classes {music, speech, scream} then centroid C is checked.

   1. IF $\mathrm{ZCR}_{\mathrm{CV}} < 20$ && $\mathrm{ZCR}_{\mathrm{Mean}} > 0.060$ && C_{CV} is < 10 (ZCR value is low, and mean value is high; centroid CV value is as low as less than 10.) → SCREAM

   If this condition does not hold, go to step 4.
4. Now two labels are left {music and speech}

   1. E_{CV} (speech) > E_{CV} (music). If $E_{\mathrm{CV}} < 100$ audio may be a music signal
   2. $\mathrm{ZCR}_{\mathrm{CV}}$, $\mathrm{ZCR}_{\mathrm{mean}}$, C_{CV} are lower for music signal than speech signal.
   3. Compare the calculated value for the audio with the vector for speech signal and music signal. The vector is represented as shown below.
      <$\mathrm{ZCR}_{\mathrm{CV}}$, $\mathrm{ZCR}_{\mathrm{Mean}}$, C_{CV}>

      $$\begin{array}{cc} \text{SPEECH SIGNAL:} & <76.30, 0.0429, 75.72> \\ \text{MUSIC SIGNAL:} & <57.89, 0.0299, 23.54> \end{array}$$

      Calculate the difference of the values from the respective vectors.

4. Percentage of **silence intervals** in speech is more than music. Speech contains a series of discontinuous unvoiced and voiced segments.

$$\mathrm{SI} = \frac{\text{Number of signal values with amplitude} < 0.01}{\text{Length of signal } (L)} \times 100$$

5. The classification of audio signal into music or speech is done by the combination of difference values of audio signal from the vectors and the silence interval.

If difference is less for music signal && SI < 3.00 → MUSIC ELSE IF,
Difference is less for the speech signal && SI > 3.00 → SPEECH,
Otherwise the audio is classified as unknown class.

Figure 6 shows that speech segment has more number of silence intervals because when a person speaks, the pauses in between the sentences or words are the silent intervals whose amplitude value is less than 0.01.

Figure 7 shows that the percentage of silent interval in a music segment is less as compared to the speech segments; the reason behind this is that music is tonal in nature. Even if the value of amplitude is less than 0.01 for a certain time frame still the duration of the frame will be smaller than speech.

Fig. 6 Silence interval in speech signal

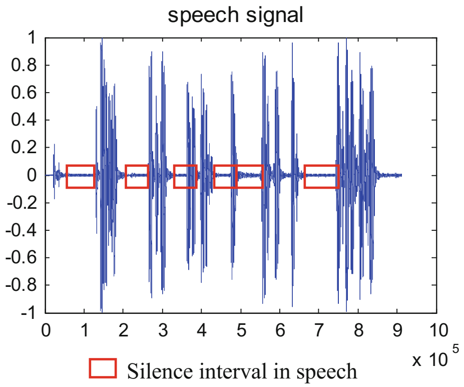
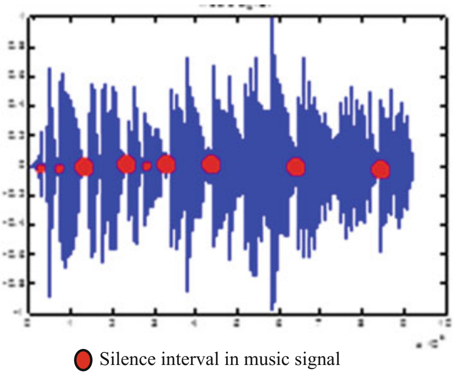


Fig. 7 Silence interval in music signal



4 Experimental Results

The classification rules are applied on the audio segments with a sampling rate of 44.1 kHz. Twenty-five different audio segments with gunshots, screams, music and speech signals are tested. Twenty-one audio samples are assigned correct labels, and 4 audio signals were incorrectly labelled. The recall rate is (21/25) * 100 = 84%.

Table 4 lists snapshot of 17 test videos out of the 25 test audio segments used for the analysis. The statistics vector for each audio is calculated. For a test video, Calculate statistics module calculates the value vector for each audio signal; a series of classification rules are applied on the vector value, and the output of the analyser is the class of the audio signal.

Table 4 Snapshot

S.No	Audio	E_{CV}	ZCR_{CV}	ZCR_{Mean}	C_{CV}	RO_{Mean}	RO_{Med}	SI	Actual class	Predicted class
1	G1	150	35.31	0.129	23	0.40	0.40	–	Gunshot	Gunshot
2	G2	103.53	47.45	0.1102	50.219	0.49	0.44	–	Gunshot	Speech
3	G3	103.46	119.99	0.1126	38.71	0.46	0.40	–	Gunshot	Speech
4	G4	111.76	31.42	0.0932	47.01	0.42	0.47	–	Gunshot	Music
5	G5	117.8	52.08	0.1039	75.04	0.47	0.42	–	Gunshot	Gunshot
6	G6	107.21	121.03	0.2042	128.16	0.48	0.43	–	Gunshot	Gunshot
7	G7	234.42	9.97	0.0619	116.33	0.46	0.41	–	Gunshot	Gunshot
8	G8	206.84	112.74	0.2394	113.20	0.50	0.45	–	Gunshot	Gunshot
9	Amy	109.45	86.16	0.0845	96.16	0.42	0.45	13.85	Speech	Speech
10	Brian	109.43	77.85	0.0886	89.77	0.41	0.44	10.79	Speech	Speech
11	Emma	106.50	94.68	0.0765	98.35	0.44	0.45	11.75	Speech	Speech
12	Joely	124.27	99.26	0.0747	98.63	0.40	0.45	10.60	Speech	Speech
13	G2	160.82	30.69	0.0732	29.36	0.39	0.42	–	Gunshot	Gunshot
14	M1	58.36	56.84	0.0330	37.69	0.40	0.41	0.1988	Music	Music
15	M2	103.86	27.98	0.0348	23.54	0.42	0.43	1.39	Music	Music
16	Scream1	161.03	9.0443	0.0667	8.2405	0.35	0.35	–	Scream	Scream
17	Scream2	125.64	15.0832	0.0705	7.5851	0.36	0.35	–	Scream	Scream

5 Conclusion

The audio-based classifier is a multiple class problem, and a number of energy-based and spectrum frequency-based features have been extracted. The time-domain features like energy, entropy, zero-crossing rate and the frequency-domain features like centroid and roll-off are used in this classification technique. A new statistics coefficient of variation has been selected and tested for the audio samples. This paper presents a list of classification rules which are designed by the analysis of the statistics values of all the features for the audio in the training dataset. In future research, features such as MFCC, chroma-based features, auto-correlation functions, pitch factors can be included to increase the efficiency of the classifier.

References

1. Giannakopoulos, T., Markis, A., Kosmopoulous, D., Perantosis, S., Theodoridis, S.: Audio-visual fusion for detecting violent scenes in videos. In: Artificial Intelligence: Theories, Models and Applications, pp. 91–100. Springer, Berlin Heidelberg (2010)
2. Zou, X., Wu, O., Wang, Q., Hu, W., Yang, J.: Multi-modal based violent movies detection in video sharing sites. In: Intelligent Science and Intelligent Data Engineering, pp. 347–355. Springer, Berlin Heidelberg (2013)
3. Pikrakis, A., Giannakopoulos, T., Theodoridis, S.: Gunshot detection in audio streams from movies by means of dynamic programming and bayesian networks. In: Acoustics, Speech and Signal Processing, 2008, ICASSP 2008, pp. 21–24. IEEE International Conference (2008)
4. Vozarikova, E., Juhar, J., Cizmar, A.: Dual shots detection. In: Advances in Electrical and Electronic Engineering, pp. 297–302 (2012)
5. Gerosa, L., Valenzise, G., Tagliasacchi, M., Antonacci, F., Sarti, A.: Scream and gunshot detection in noisy environments. In: 15th European Signal Processing Conference (EUSIPCO-07), 3–7 Sep, Poznan, Poland (2007)
6. Giannakopoulos, T., Kosmopoulos, D., Aristidou, A., Theodoridis, S.: Violence content classification using audio features. In: Advances in Artificial Intelligence 4th Helenic Conference on AI, SETN 2006, Heraklion, Crete, Greece (2006)

Document-to-Sentence Level Technique for Novelty Detection

Sushil Kumar and Komal Kumar Bhatia

Abstract Novelty identification is accustomed to distinguishing novel data from an approaching stream of documents. In this study, we proposed a novel methodology for document-level novelty identification by utilizing document-to-sentence-level strategy. This work first splits a document into sentences, decides the novelty of every sentence, then registers the record-level novelty score in view of an altered limit. Exploratory results on an arrangement of document demonstrate that our methodology beats standard document-level novelty discovery as far as repetition exactness and excess review. This work applies on the document-level information from an arrangement of documents. It is valuable in identifying novel data in information with a high rate of new documents. It has been effectively incorporated in a true novelty identification framework in the zone of information retrieval.

Keywords Novelty identification · Sentence segmentation · Document novelty identification

1 Introduction

There is a nonstop increment in the information content that is transferred through the Internet between customers, administrations, and Internet clients [1]. Individuals who are in media, security offices get an immense measure of stories, papers, articles, and reports from an expansive number of resources. Such troublesome circumstance propelled the scientists to concoct new programmed framework which is in view of novelty identification. The most recent decade saw an expanding enthusiasm for the novelty location which expects to manufacture programmed

S. Kumar (✉) · K. K. Bhatia
YMCA, University of Science and Technology, Faridabad 121006, India
e-mail: panwar_sushil2k@yahoo.co.in

K. K. Bhatia
e-mail: komal_bhatia1@rediffmail.com

S. S. Agrawal et al. (eds.), *Speech and Language Processing for Human-Machine Communications*, Advances in Intelligent Systems and Computing 664,
https://doi.org/10.1007/978-981-10-6626-9_2

frameworks which are proficient to disregard previous stories, papers, and articles as of now read or known and tell the clients of such frameworks about any new stories, papers, reports, and articles. There is an expanding requirement for distinguishing novel and important data out of a mass of approaching content reports. Novel data for this situation alludes the message which contain new substance and novelty recognition is the procedure of singling out novel data [2–4] from a given arrangement of content documents [5]. Thus due to this procedure, clients can spare time by perusing just the new data, while the rehashed data is separated out.

2 Literature Review

Event level and sentence level are two ways for novelty identification. We provide a review for novelty identification in brief by the related research as follow.

2.1 Novelty Identification Using Event Level

This work is based on online new event identification [3, 6–13]. Available techniques for new event identification are related to clustering algorithms.

2.2 Novelty Identification Using Sentence Level

Study on novelty identification at the sentence level is identified with the TREC novelty tracks [3, 14–16]. Different exploration gatherings provide an interest in the TREC 2002–2004 novelty tracks and reported their outcomes [2].

2.3 Comparison

In this study, a new approach for novelty identification at document level has been proposed that is different from the available approaches in the literature in the following sense:

(a) Available approaches assume sentences and documents as two different resources and decide novelty individually.
(b) The proposed approach regards a document as redundant if it shares a single sentence with the history document.
(c) The proposed work mainly focuses on sentence-level module, which, in turn, improves code reuse for novelty identification.

3 Proposed Work for Novelty Detection at Document Level

The idea of novelty detection will optimize the search engine results. Many applications have utilized novelty identification to reduce nonredundant information presented to user. In this study, a novel approach to document level has been proposed. The algorithm is accustomed to remove the redundancy of the results, which increases speed to find novel information in the documents. A novelty score is calculated by sentence segmentation instead of whole document. The document is required to be preprocessed for sentence segmentation [17].

3.1 Document-Level Novelty Detection Algorithm

Document-level novelty detection (DND) algorithm is a proposed detection algorithm which is used to find whether a document provides a novel information or not, when compared with the history documents. DND first splits the document into sentences [17] and computes the novelty score of each document based on a fixed threshold. Sentence segmentation is used a tool name Stanford parser, which splits the document into sentences. Sentences are then compared with all the history sentences to compute the similarity between those sentences.

To compute the nature of document, similarity is converted to novelty score for each sentence. A minimum value is selected out of the novelty values and finally, the decision has to be made. The architecture of the proposed system is show in Fig. 1.

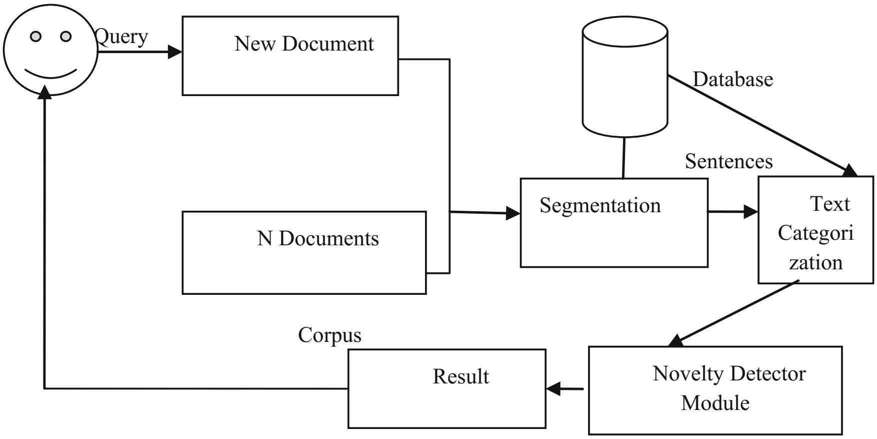


Fig. 1 Architecture of proposed system

3.2 Novelty Detector Module

This module helps in discovering the document novelty. The procedure of this module is as demonstrated in Fig. 2. The document is divided into sentences; register the novelty score of every sentence by utilizing the sentence-level novelty identification. At that point, the normal of novelty score is compared with threshold, and if the estimation of novelty score is more prominent than the fixed threshold, then the document is considered as novel generally not.

For similarity measure, cosine similarity is used for good performance to identify the novel information between sentences. This has been cleared form the existing study. The cosine similarity is defined between two sentences as:

$$\cos(s_t, s_i) = \frac{\sum \mathrm{wk}(s_t)\mathrm{wk}(s_i)}{\|s_t\| \cdot \|s_i\|}$$

The novelty similarity score is calculated as (1-cosine similarity score). Each of the history sentences is separately compared with the current sentence. Then minimum novelty score from them provides the novelty score of the current sentence [18].

4 Experimental Result Analysis

The proposed architecture has been simulated by using an example. User chooses a new document and that document is compared with three documents. The result is computed by finding the novelty score for each document based on a fixed threshold.

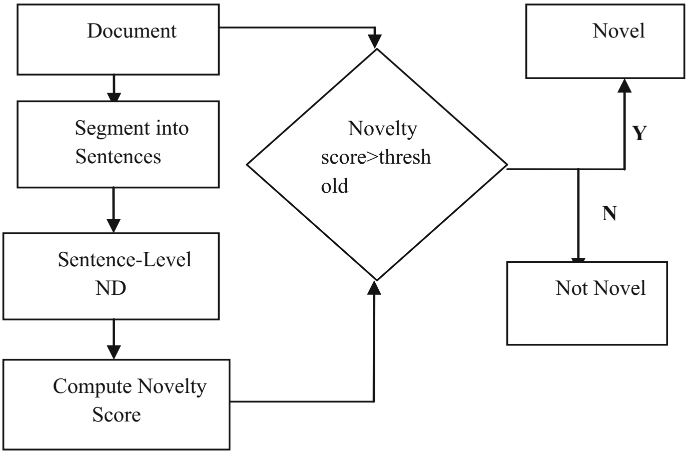


Fig. 2 Process of novelty detection

Example

Step 1 Three documents (N1, N2, and N3) have been taken for the basic analysis. N2 and N3 documents did not show here, but the calculation is performed on these documents for results comparison (Fig. 3)

Step 2 Now user selects a new document (newDoc) (Fig. 4)

Step 3 All the documents are segmented into sentences, and each sentence of the new document are compared with all the sentences of history documents

Step 4 Sentences of N1 document are taken one by one

Step 5 Len (newDoc1) == Len (senN1)

Step 6 Find cosine similarity of each sentence (applied the same for N2 and N3) (Table 1)

Step 7 Now the maximum values of cosine similarity from each table is selected (Table 2)

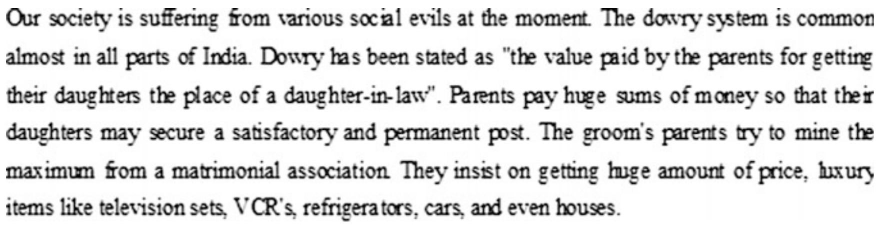
Step 8 Find novelty score for each document (Table 3)

Step 9 Now compute minimum novelty score for each document (Table 4)

Step 10 Find the average novelty score
avgNovel = (0.15 + 0.10 + 0.02)/3 = 0.09

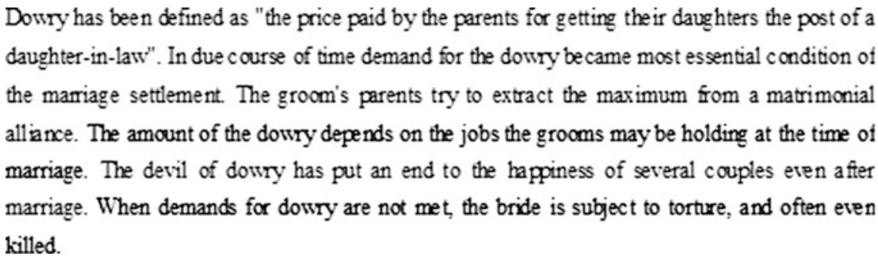
Step 11 Now we compare the average novelty score with the fixed threshold value Threshold = 0.45
avgNovel = 0.09 which is less than the threshold value

So, new document ND is not novel.



Our society is suffering from various social evils at the moment. The dowry system is common almost in all parts of India. Dowry has been stated as "the value paid by the parents for getting their daughters the place of a daughter-in-law". Parents pay huge sums of money so that their daughters may secure a satisfactory and permanent post. The groom's parents try to mine the maximum from a matrimonial association. They insist on getting huge amount of price, luxury items like television sets, VCR's, refrigerators, cars, and even houses.

Fig. 3 Document N1



Dowry has been defined as "the price paid by the parents for getting their daughters the post of a daughter-in-law". In due course of time demand for the dowry became most essential condition of the marriage settlement. The groom's parents try to extract the maximum from a matrimonial alliance. The amount of the dowry depends on the jobs the grooms may be holding at the time of marriage. The devil of dowry has put an end to the happiness of several couples even after marriage. When demands for dowry are not met, the bride is subject to torture, and often even killed.

Fig. 4 New document

Table 1 Cosine similarity values of N1 document

ND1	0.842	0.389
ND2	0.428	0.11
ND3	0.85	–
ND4	0.127	–
ND5	0.427	0.117
ND6	0.252	0.06

Table 2 Maximum values from each table

	N1	N2	N3
ND1	0.84	0.33	0.13
ND2	0.43	0.90	0.16
ND3	0.85	0.23	0
ND4	0.13	0.59	0.98
ND5	0.43	0.86	0.28
ND6	0.25	0.26	0.88

Table 3 Novelty scores

New document	N1	N2	N3
ND1	0.16	0.67	0.87
ND2	0.57	0.10	0.84
ND3	0.15	0.77	1
ND4	0.87	0.41	0.02
ND5	0.57	0.14	0.72
ND6	0.75	0.74	0.12

Table 4 Minimum novelty scores

ND	N1	N2	N3
	0.15	0.10	0.02

From the result analysis, it has been proved that this proposed method provides proper result in lesser amount of time and with better efficiency.

5 Conclusion

In this paper, a system has been suggested that aptly applies document-level novelty identification. This structure makes record-level novelty identification more powerful by receiving the procedures for the sentence level. Results demonstrate that proposed strategy significantly enhances the document-level novelty identification execution as far as repetition accuracy and excess review [19–22]. The perceptions are exceptionally useful for effectively coordinating DND to a true novelty identification framework in information processing [23–32].

References

1. Greengrass, E.: Information Retrieval: A Survey, DOD Technical Report TR-R52-008-001 (2000)
2. Salton, G., McGill, M.J.: Introduction to Modern Information Retrieval. McGraw-Hill, ISBN 0070544840 (1983)
3. ciir.cs.umass.edu
4. www.sersc.org
5. Soboroff, I., Harman, D.: Novelty detection: the TREC experience. In: Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing, Vancouver, British Columbia, Canada, pp. 105–112 (2005)
6. Allan, J., Wade, C., Bolivar, A.: Retrieval and novelty detection at the sentence level. In: Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Toronto, Canada, pp. 314–321 (2003)
7. Ng, K.W., Tsai, F.S., Chen, L., Goh, K.C.: Novelty detection for text documents using named entity recognition. In: 2007 6th International Conference On Information, Communications And Signal Processing, ICICS (2007)
8. Allan, J., Paka, R., Lavrenko, V.: On-line new event detection and tracking. In: Proceedings of SIGIR-98, pp. 37–45 (1998)
9. Yang, Y., Zhang, J., Carbonell, J., Jin, C.: Topic-conditioned novelty detection. In: SIGKDD, pp. 688–693 (2002)
10. Stokes, N., Carthy, J.: First story detection using a composite document representation. In: Proceedings of HLT01 (2001)
11. Franz, M., Ittycheriah, A., McCarley, J.S., Ward, T.: First story detection, combining similarity and novelty based approach. Topic Detection and Tracking Workshop (2001)
12. Allan, J., Lavrenko, V., Jin, H.: First story detection in TDT is hard. In: Proceedings of CIKM (2000)
13. Yang, Y., Pierce, T., Carbonell, J.: A study on retrospective and on-line event detection. In: Proceedings of SIGIR-98 (1198)
14. Brants, T., Chen, F., Farahat, A.: A system for new event detection. In: Proceedings of ACM SIGIR2003 (2003)
15. Kumaran, G., Allan, J.: Text classification and named entities for new event detection. In: Proceedings of ACM SIGIR, pp. 297–304 (2004)
16. Harman, D.: Overview of the TREC 2002 Novelty Track. In: TREC (2002)
17. Tsai, F.S.: D2S: document-to-sentence framework for novelty detection. Knowl. Inf. Syst. (2010)
18. Verhaegen, P.-A., Vandevenne, D., Duflou, J.R.: Originality and novelty: a different universe. In: Proceedings of DESIGN 2012, the 12th International Design Conference, Dubrovnik, Croatia, pp. 1961–1966 (2012)
19. Brants, T., Chen, F., Farahat, A.: A system for new event detection. In: Proceedings of SIGIR-03, pp. 330–337 (2003)
20. Soboroff, I., Harman, D.: Overview of the TREC 2003 novelty track. In: TREC (2003)
21. Soboroff, I.: Overview of the TREC 2004 novelty track. In: TREC (2004)
22. Allan, J., Bolivar, A., Wade, C.: Retrieval and novelty detection at the sentence level. In: Proceedings of SIGIR-03 (2003)
23. Kazawa, H., Hirao, T., Isozaki, H., Maeda, E.: A machine learning approach for QA and novelty tracks: NTT system description. In: TREC-10 (2003)

24. Qi, H., Otterbacher, J., Winkel, A., Radev, D.T.: The University of Michigan at TREC2002: question answering and novelty tracks. In: TREC (2002)
25. Eichmann, D., Srinivasan, P.: Novel results and some answers, The University of Iowa TREC-11 results. In: TREC (2002)
26. Zhang, M., Song, R., Lin, C., Ma, S., Jiang, Z., Jin, Y., Zhao, L.: Expansion-based technologies in finding relevant and new information: THU TREC2002 novelty track experiments. In: TREC (2002)
27. Kwok, K.L., Deng, P., Dinstl, N., Chan, M.: TREC2002, novelty and filtering track experiments using PRICS. In: TREC (2002)
28. Zhang, Y., Callan, J., Minka, T.: Novelty and redundancy detection in adaptive filtering. In: Proceedings of SIGIR (2002)
29. Tsai, M., Hsu, M., Chen, H.: Approach of information retrieval with reference corpus to novelty detection. In: TREC (2003)
30. Jin, Q., Zhao, J., Xu, B.: NLPR at TREC 2003: novelty and robust. In: TREC (2003)
31. Sun, J., Yang, J., Pan, W., Zhang, H., Wang, B., Cheng, X.: TREC-2003 novelty and web track at ICT. In: TREC (2003)
32. Litkowski, K.C.: Use of metadata for question answering and novelty tasks. In: TREC (2003)

Continuous Hindi Speech Recognition in Real Time Using NI LabVIEW

Ishita Bahal, Ankit Mishra and Shabana Urooj

Abstract Speech is the most common form of communication, and the need of the hour is a robust speech recognition system. This paper aims to present an algorithm to design a continuous speech recognition system. The recognition of the speech utterances is done on a real-time basis using NI LabVIEW.

Keywords Continuous speech recognition · ASR · LabVIEW
Real time · MFCC

1 Introduction

Speech recognition application areas may have to contend with a noisy environment. This calls for processing techniques that should be little affected by background noise and therefore on the performance of the recognizer. The human auditory system is robust to background noise, so it becomes a necessity to have a speech recognizer with robust performance.

A speech recognition system requires automatic speech recognition (ASR) capabilities. An ASR system is subjected to fluctuating speech signals.

The major reasons for fluctuation in speech are (a) acoustic media; (b) inter-speaker variability; (c) intra-speaker variability [1].

I. Bahal (✉) · A. Mishra · S. Urooj
Department of Electrical Engineering, School of Engineering,
Gautam Buddha University, Greater Noida, Uttar Pradesh, India
e-mail: ishita.bahal@yahoo.in

A. Mishra
e-mail: ankitmishra723@gmail.com

S. S. Agrawal et al. (eds.), *Speech and Language Processing for Human-Machine Communications*, Advances in Intelligent Systems and Computing 664,
https://doi.org/10.1007/978-981-10-6626-9_3

2 Automatic Speech Recognition System

Human speech recognition endures under numerous sorts of antagonistic conditions. Replication of this execution is a definitive objective in ASR research. In order to accomplish such execution, the structure of the ASR framework ought to be designed according to the human auditory system.

The learning that we have about both the human auditory system and speech production mechanism impact most feature representation. Some of these highlight representations utilize just some straightforward ideas [2, 3], while different representations endeavor to reenact the auditory system [4–6].

It is widely acknowledged that for speech recognition the phase spectrum is disregarded in light of the fact that in the standard ASR framework, speech is processed using small temporal window durations of 20–40 ms [7] and at such window lengths the magnitude spectrum provides more intelligibility when contrasted with the phase spectrum [8–10].

3 Creating a Continuous Recognition System

In designing a continuous speech recognition system, the following steps are involved in the process:

Step 1: Acquisition of Data

Acquisition of speech data is done using the Acquire Sound Express VI available in the functions pallet; the VI uses available devices inbuilt or connected to the system.

In this project, a laptop inbuilt mike "Microphone (PnP sound device)" is used as the acquisition transducer for speech data.

16-bit resolution is taken for the digitization of the speech data. The number of quantization levels will therefore be 2^{16} = 65,536 and is found to be optimal for preserving information present in the analog version of the speech signal.

The sample rate is taken as 11,025 Hz, this value is the lowest sample rate value that the device supports.

The Express VI is made to acquire data for total duration of approximately 3 s, as it takes almost 1–1.5 s for uttering a word.

Step 2: Preprocessing

At this stage, the signal is passed through a FIR filter to discard the speaker-specific characteristics and environment-specific characteristics. Filter coefficients are 1 and −0.95. From the filtered signal, we calculate the energy of the signal as:

$$E = T \sum x^2.$$

The filtered signal is now framed into smaller frames and passed through the Hamming window given by:

$$w(n) = 0.54 - 0.46\cos\frac{2n\pi}{N}.$$

There are a number of window functions possible such as rectangular, triangular, Hanning, Hamming, Blackman. Hamming window is a good choice because it has the smallest side lobe magnitude for a given main lobe width [11] (Fig. 1).

The filtered signal is now framed into smaller frames and passed through the Hamming window given by:

$$w(n) = 0.54 - 0.46\cos\frac{2n\pi}{N}.$$

Output from the Hamming window gives us the frames in which the word is spoken.

Frame length is determined as: dt/t_o; for a 50 Hz desired frequency, t_o = 0.02 (Fig. 2).

Step 3: Mel Frequency Cepstral Coefficients

Mel Frequency Cepstral Coefficients (MFCCs) are widely used for speech recognition as they have been found to best represent the human auditory system.

The Mel scale can be approximated by the following equation [1]:

$$M(f) = 2595\log 10(1 + f/700),$$

where

f	is the linear frequency (in Hertz);
$M(f)$	is the perceived frequency (in Mel).

The Mel filter bank created is applied to the power spectrum of each frame (Fig. 3).

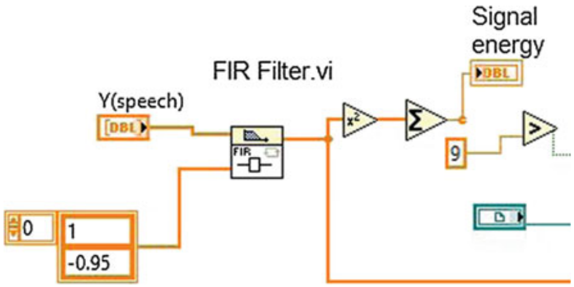


Fig. 1 Speech data passed through the FIR filter

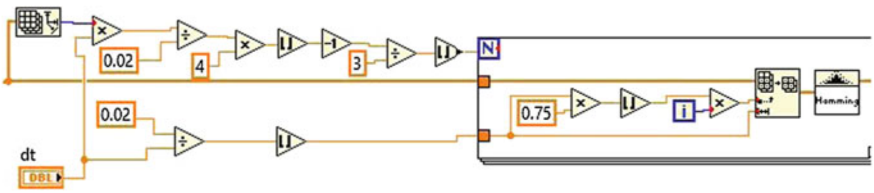


Fig. 2 Windowing of the speech data

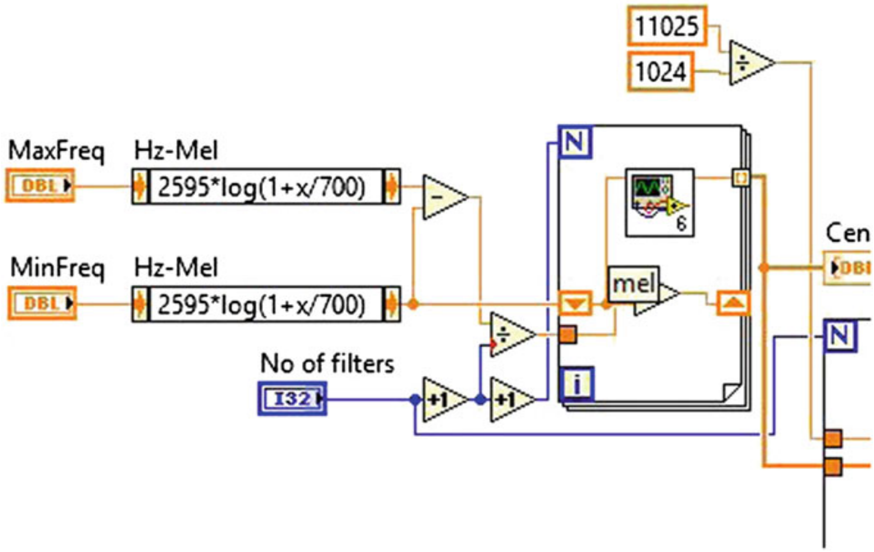


Fig. 3 Mel filter bank

Fast Fourier transform of each word frames obtained above is taken, and FFT being a faster implementation of the DFT reduces an *N*-point Fourier transform to (*N*/2 log 2*N*).

With an FFT you can only evaluate a fixed length waveform containing 512 points, or 1024 points, or 2048 points, etc. therefore a 1024—FFT gives 512 point Spectrum (Fig. 4).

The log magnitude of the filter bank energy is calculated to factor in the natural log compression of the auditory system. Next, a discrete cosine transform is applied to these log energies to de-correlate the data [12].

The MFCCs were extracted from a frame of 20 ms with an overlapping of 10 ms. Thirteen cepstral coefficients were extracted and fed to classifiers for recognition (Fig. 5).

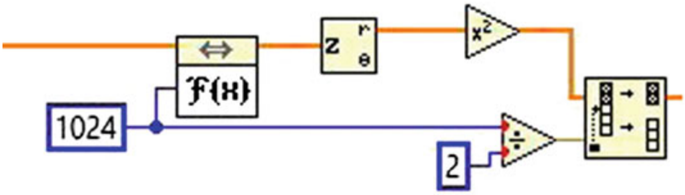


Fig. 4 FFT of the speech data

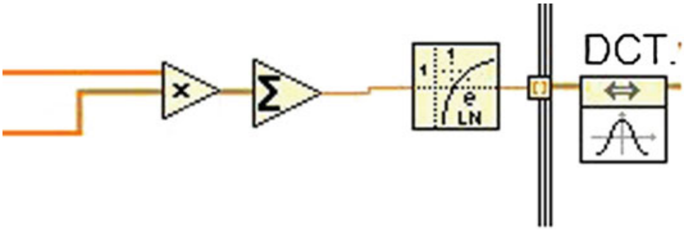


Fig. 5 MFCC generation, log of filter bank energies and applying it to DCT

The mean of the outputs obtained from DCT gives us the speech utterance vector, which is first used for creating a database of each word and later used for classification by matching it with saved vector.

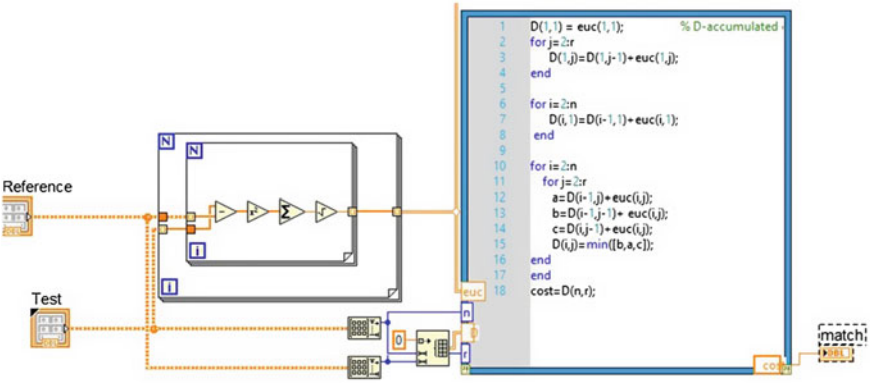
4 Database Creation

The speech utterance vector is obtained 3 times by acquiring the speech data separately every time, and saved for each word, this data can be exported with a TDMS format and read later.

5 Classification

The input word is sent to a DTW VI, and this DTW is an algorithm used for measuring similarity between two temporal sequences which vary in time and speed and used as distance measure for “nearest neighbor classifier” (Fig. 6).

The algorithm is now used to match the test value and the value saved for training, thus a distance array is obtained. The minimum distance obtained by comparing the test value to saved values gives us the approximation of best match value (Fig. 7).



```
D(1,1) = euc(1,1);                % D-accumulated
for j=2:r
    D(1,j)=D(1,j-1)+euc(1,j);
end

for i=2:n
    D(i,1)=D(i-1,1)+euc(i,1);
 end

for i=2:n
  for j=2:r
    a=D(i-1,j)+euc(i,j);
    b=D(i-1,j-1)+ euc(i,j);
    c=D(i,j-1)+euc(i,j);
    D(i,j)=min([b,a,c]);
end
end
cost=D(n,r);
```

Fig. 6 DTW algorithm

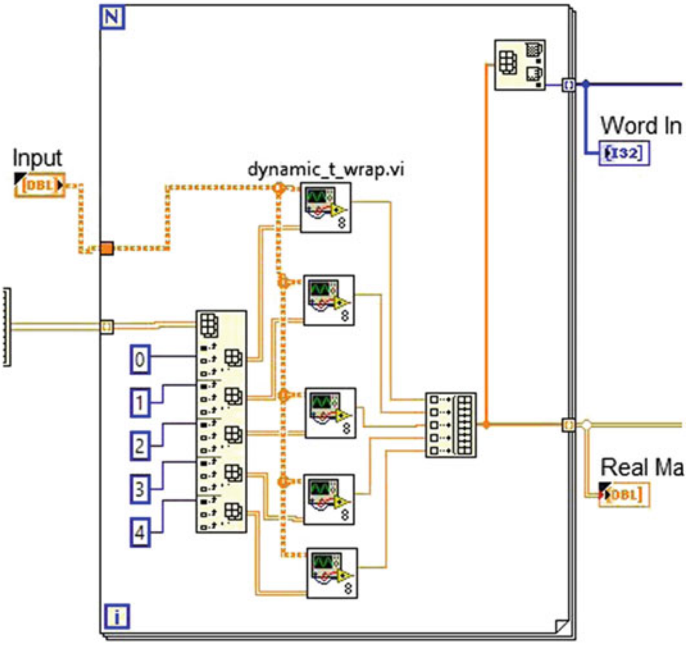


Fig. 7 Calculating the nearest neighbor

6 Result

The main objective of this work was to design a continuous speech recognition system, which acquires data in real time and on the basis of the classification of its acoustic vector a control signal value is generated, thus indicating whether the input speech was correctly recognized.

References

1. Bahal, I., Urooj, S.: Hindi speech recognition using NI LabVIEW. In: Proceedings of the International Conference on Futuristic Trends in Computational Analysis and Knowledge Management, IEEE Xplore Digital Library, India (2015)
2. Davis, S.B., Mermelstein, P.: Comparison of parametric representations for mono-syllabic word recognition in continuously spoken utterances. IEEE Trans. Acoust. Speech Signal Process. **28**(4), 357–366 (1980)
3. Hermansky, H.: Perceptual linear predictive (PLP) analysis of speech. J. Acoust. Soc. Am. **87** (4), 1738–1752 (1990)
4. Ghitza, O.: Auditory models and human performance in tasks related to speech coding and speech recognition. IEEE Trans. Speech Audio Process. **2**(1), 115–132 (1994)
5. Kim, D., Lee, S., Kil, R.M.: Auditory processing of speech signals for robust speech recognition in real-world noisy environments. IEEE Trans. Speech Audio Process. **7**(1), 55–69 (1999)
6. Seneff, S.: A joint synchrony/mean-rate model of auditory speech processing. J. Phonetics **16**, 55–76 (1998)
7. Paliwal, K.K., Alsteris, L.D.: On the usefulness of STFT phase spectrum in human listening tests. Speech Commun. **45**(2), 153–170 (2005)
8. Schroeder, M.R.: Models of hearing. In: Proceedings of the IEEE, vol. 63, pp. 1332–1350 (1975)
9. Oppenheim, A.V., Lim, J.S.: The importance of phase in signals. In: Proceedings of the IEEE, May 1981, vol. 69, pp. 529–541 (1981)
10. Liu, L., He, J., Palm, G.: Effects of phase on the perception of intervocalic stop consonants. Speech Commun. **22**(4), 403–417 (1997)
11. Lathi, B.P.: Signal Processing and Linear Systems. USA Berkeley Cambridge Press, California (1998)
12. Farooq, O., Datta, S.: Speech recognition with emphasis on wavelet based feature extraction. IETE J. Res. (2002)

Gujarati Braille Text Recognition: A Design Approach

Hardik Vyas and Paresh Virparia

Abstract In the era of technological enhancement, a very moderate amount of development is observed in the Gujarati Braille language. As visually disabled people are also an important part of the society. So in this paper, we have focused on providing design approach that recognizes Gujarati Braille Text. As, if once the Gujarati Braille Text is recognized, it can be converted into Gujarati language; so that it can be used by the sighted people and it reduces the written communication gap between blind and sighted people. We have provided various techniques that describe as a literature to develop the tool and also identified the challenges that might be faced at the time of recognition and conversion. The technique is explained by keeping in mind the perception of the standard Braille system. It would also provide new thoughts for the assistance to the visually disabled people.

Keywords Gujarati text · Braille language · Natural language processing · Pattern recognition

1 Introduction

In our daily activities, language is a mode of communication; it can be either in written or verbal form, and it is used to express our views and feelings. People with visual disabilities are also an essential part of the society. However, due to their disability they have less access to the computer, digital documents, and internet than the people with clear vision. Therefore, it is important to provide them a

H. Vyas (✉)
Babu Madhav Institute of Information Technology, Uka Tarsadia University,
Maliba Campus, Gopal Vidyanagar, Tarsadi-Bardoli, Surat, Gujarat, India
e-mail: hardikv@acm.org

P. Virparia
G. H. Patel Post Graduate Department of Computer Science and Technology,
Sardar Patel University, Vallabh Vidyanagar, Anand, Gujarat, India
e-mail: pvvirparia@yahoo.com

S. S. Agrawal et al. (eds.), *Speech and Language Processing for Human-Machine Communications*, Advances in Intelligent Systems and Computing 664,
https://doi.org/10.1007/978-981-10-6626-9_4

support through which they can communicate and interact with each other and with the sighted people. Braille [1] is a language used for written communication by visually disabled people. As visually impaired people use Braille as a medium of communication, but non-blind people are not able to understand it. So, we are trying to develop the system, which assists them to reduce written communication gap with the non-blind people. The people who are in touch with blind people or work with them and are not able to read Braille will be benefitted from the system.

Natural language processing (NLP) and text processing are an important field of research nowadays. Natural language processing (NLP) [2] is a branch of artificial intelligence that deals with analyzing, understanding, and generating the languages that humans use naturally in order to interface with computers in both written and spoken contexts using natural human languages instead of computer languages. It is also known as computational linguistics.

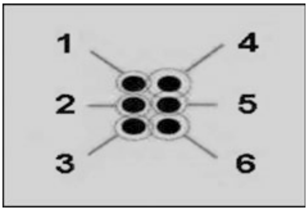
Text processing [3] is a method, i.e., transformation of text from one format to another format or one language to another language. It is also refer as the use of a computer to produce, change, and store pieces of writing.

1.1 About Braille Script

Frenchman Louis Braille [4–8] is the inventor of Braille script. He became blind due to his childhood accident. So at the age of 15, in 1824 [4], Braille developed his own code for French alphabet. In 1829, he published his system, which also contains musical notation. Visually impaired people use Braille, a series of raised dots in embossed form to read and write through touch of their fingers.

Standard Braille approach consists of Braille cells which are made up of raised dots on thick sheet of paper. These raised dots are generated through the processing of embossing. A cell consists of six dots which are arranged in the form of a rectangular grid, i.e., two dots horizontally (row) and three dots vertically (column). Due to this arrangement of dots, total sixty four different patterns of dots can be obtained. The cell will be consists of at least one raised dot and a maximum of six [9]. If in a cell no raised dots are present then it is considered as a space. (The layout of Braille cell is shown in Fig. 1 [10].)

Fig. 1 A Braille cell



A standard printed Braille sheet generally contains twenty-five rows with forty cells in each row. The approximate dimension of a standard Braille sheet is 11 inches by 11 inches. (The dimension of a Braille cell is shown in Fig. 2 [11].)

Braille documents can be printed single sided as well as double sided. When texts are printed double sided, two types of dots are formed, recto-verso. All dots on a Braille page should fall on an orthogonal grid. When texts are printed double sided (Inter-point), the grid of the inter-point text is shifted so that the dots fall in between the primary side dots (Recto-Verso dots are shown in Fig. 3 [12]).

Unified Braille script used for writing the languages of India is referred to as Bharati Braille or Bhartiya Braille. When India gained independence, eleven Braille scripts were in use in different parts of the country and for different languages. Gujarati Braille is one of the Bharati Braille. (Gujarati alphabet [13] as written in Braille is shown in Fig. 4 and punctuations [14] are shown in Fig. 5.)

1.2 About Gujarati Script

Gujarati script is derived from Devanagari script and is descended from Sanskrit. There are over 50 million people worldwide who use Gujarati for writing and speaking. The earliest known document in Gujarati script is a manuscript dating from 1592, and the script first appeared in print in 1797 advertisement. The shapes of Gujarati characters are also very typical.

The Gujarati alphabet has overall 75 distinct legitimate and recognized shapes, which mainly includes 59 characters and 16 diacritics (Gujarati letters and numerals

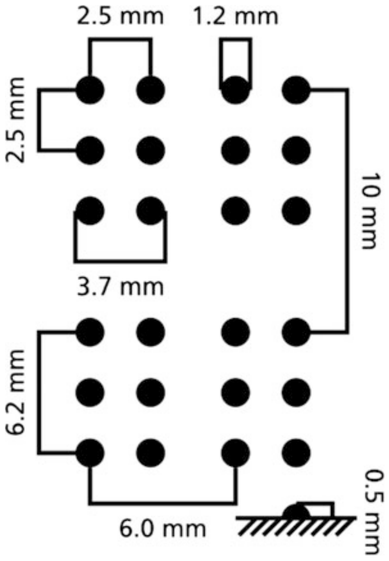


Fig. 2 A Braille cell dimension (in millimeters)

Fig. 3 Inter-point Braille

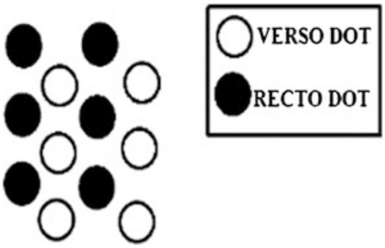


Fig. 4 Gujarati Braille

અ	આ	ઇ	ઈ	ઉ	ઊ	એ	ઐ	ઓ	ઔ
⠁	⠜	⠊	⠔	⠥	⠳	⠑	⠌	⠕	⠪
ક	ખ	ગ	ઘ	ઙ	ચ	છ	જ	ઝ	ઞ
⠅	⠨	⠛	⠣	⠬	⠉	⠡	⠚	⠴	⠒
ટ	ઠ	ડ	ઢ	ણ	ત	થ	દ	ધ	ન
⠾	⠺	⠫	⠿	⠼	⠞	⠹	⠙	⠮	⠝
પ	ફ	બ	ભ	મ	ય	ર	લ	વ	ળ
⠏	⠖	⠃	⠘	⠍	⠽	⠗	⠇	⠧	⠸
શ	ષ	સ	હ	ક્ષ	જ્ઞ	ઋ			
⠩	⠯	⠎	⠓	⠟	⠱	⠐⠗	⠻	⠐⠻	⠐⠣
અં	અઃ	અઁ	્	ઍ					
⠰	⠠	⠄	⠈	⠭	⠢	⠂			

Symbol	Parentheses		Brackets		single quote		
	open	close	open	close	open	close	?
Braille	○ ○ ● ○ ● ○ ○ ●	○ ● ○ ● ○ ○ ● ○	○ ○ ● ○ ● ● ● ●	○ ○ ○ ● ● ● ● ●	○ ○ ○ ○ ● ○ ● ●	○ ○ ○ ○ ● ○ ○ ○	○ ○ ● ● ● ● ○ ○

Fig. 5 Punctuations written in Braille

are shown in Fig. 6). Fifty-nine characters are divided into 36 consonants (34 singular and 2 compound (not lexically though)) means ornamented sounds, 13 vowels (pure sounds), and 10 numerical digits [15]. Sixteen diacritics are divided into 13 vowel and 3 other characters. The alphabet is ordered by logically grouping of the vowels and the consonants based on their pronunciations [16].

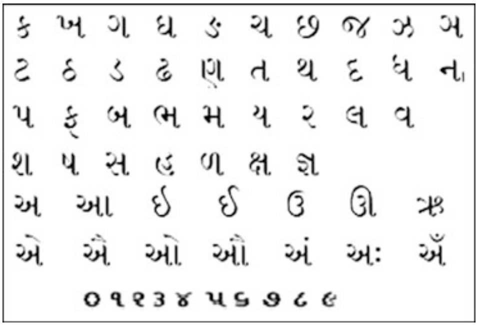
Gujarati script is written from left to right. The vowels are called Swar and consonants are called Vyanjan. Gujarati consists of set of special modifier symbols called Maatras, corresponding to each vowel, which are attached to consonants to change their sound. Modifiers are placed at the top, at bottom right, or at bottom part of consonant. They are attached at different positions for different consonants. They can also occur in different shapes. If two half characters are joined then it is called a conjunct [17].

2 Literature Review

Following are some of the research work done in Braille to text conversion. It includes brief overview of different character recognition techniques used by the researchers and also what experimental result they achieved during their studied:

Mousa et al. [18] have proposed a system to recognize characters for a single-sided Braille document. Their system works with all size of the scanned Braille image. The system uses various image processing phases such as image acquisition, image preprocessing for noise removal, segmentation of modified image, feature extraction, and then character recognition. They have enhanced each stage to develop good quality system. Experimental result shows that system achieves recognition accuracy of 94.39%.

Rajarapollu et al. [10] have developed a system for the conversion of Braille to text and speech of English language based on field programmable gate array (FPGA) Spartan3 kit. To provide input to the system Braille keypad is designed that consists of different combinations of cells. The input goes to the FPGA Spartan3 kit and according to the input, depending of the decoding logic in VHDL (VHSIC (Very High Speed Integrated Circuit) Hardware Description Language) language FPGA converts the input into equivalent English text. Finally, with the help of algorithm the text is also converted to speech.



ક ખ ગ ઘ ઙ ચ છ જ ઝ ઞ
ટ ઠ ડ ઢ ણ ત થ દ ધ ન
પ ફ બ ભ મ ય ર લ વ
શ ષ સ હ ળ ક્ષ જ્ઞ
અ આ ઇ ઈ ઉ ઊ ઋ
એ ઐ ઓ ઔ અં અઃ અઁ
૦ ૧ ૨ ૩ ૪ ૫ ૬ ૭ ૮ ૯

Fig. 6 Gujarati characters and digits

Padmavathi et al. [19] proposed a method that converts a scanned Braille document to text (English, Hindi, and Tamil language) that can be read out at latter stage through the computer. Preprocessing is performed on the Braille documents to reduce the noise and enhance the dots. Segmentation of Braille cells is done and dots are extracted from the cells which are then converted to the number sequence. Then the number sequence is mapped to the suitable letter of the language. A speech synthesizer is used to speak out the converted text. The Braille characters can also be typed with the help of number pad of the keyboard which can also be mapped to the letter of the language and spoken out.

Shreekanth and Udayashankara [12] present an algorithmic approach for the recognition of double-sided embossed Braille document. They have used Braille dot analysis which is based on the variation in the gray level values of the Braille image. The difference in gray level is due to the projections and dejections created on the document. They have not only recognized the Braille dots but also detected recto and verso dots from inter-point Braille document. They have used thresholding, centroid detection, mask design, placement of designed mask on the centroid detected dots. The average processing time taken for processing is 5.6 s. Authors have experimented on the developed database and obtained 99% of recognition rate.

Authman and Jebr [20] describe a method for recognizing Braille cells in single-sided Arabic Braille document. To binaries the Braille image is the challenging task of Optical Braille Recognition, but they have found the solution to it. They used algorithms based on Morphological operations to binaries Braille documents (i.e., morphological top hat filter on green Braille documents and morphological bothat filter in yellow Braille documents). The system recognizes printed Braille cells and converts them to the regular text based on the ASCII code of the Arabic letter. To recognize the characters they have used template matching technique. The execution time taken by the system was 8–38 s. They got accuracy of 98.04–100% for green Braille documents and 97.08–99.65% for yellow Braille documents.

Wong et al. [21] proposed a solution to recognize single-sided embossed optical Braille documents. They have used image processing algorithm and probabilistic neural network. They have generated the output by preversing the layout of the original document that can again be printed on the Braille Printer. They got recognition rate of 99%.

Mennens et al. [22] have developed a system that converts Braille image to the plain text. They have predefined their constraints on which the system would work. So once the text is generated it can be reproduced with the help of embosser.

Al-Saleh et al. [23] have presented an algorithm to detect Braille dots from the scanned image of embossed Braille document. They used mixture of Beta distributions with thresholding to obtain the histogram of the Braille document image. Then a grid is formed of recto and verso dots from segmented Braille image. Through the proposed technique they got good results.

Al-Salman and Fathi [24] have developed a system using MATLAB environment for Optical Arabic Braille Recognition and convert it to text and voice. Image processing technique is used which performs binary conversion, edge detection, holes filling, and image filtering on Braille document before the extraction of the dot. Comparison of the Braille dot position in each cell is done with the database. They have also generated unique decimal code for each Braille cell used for the reconstruction of word reconstruction according to the voice and text conversion database. The algorithm achieved expected result for the recognition of letter and words. The transcription accuracy obtained is over 99% and average processing time is 32.6 s/page.

Blenkhorn [7] describes a method for converting Braille to text that can be stored in the computer, so that it can be printed at the latter stage. The algorithm is developed by considering Standard English Braille. It used table driven method for the conversion. The system can also be configured, so that it can be used for many different languages and character sets.

3 Challenges

Braille is a language that is made up of cells to represents characters (Braille cell is shown in Fig. 2). There are total 6 dots in single cell. So, total 64 characters can be written. But Gujarati language has total 75 characters. So in Gujarati Braille (shown in Fig. 4), there are some characters (consonant and vowels) that are identical to the numerals when represented in Braille. So some assumptions are considered while writing Gujarati Braille.

- Digits 0–9 are represented in the same way as some consonants and vowels, i.e., ૧ is represented in the same way as, અ, ૨ is represented in the same way as બ and so on. So, digit identifier # character (Dots 3, 4, 5, 6) is used to indicate that the character written is digit and not a consonant or vowels.
- If digits and letters are to be written in a single word, then to separate digit from the letter, letter sign is used before the letters starts in the word. It indicates that the characters written after the letter sign are not digits but letters (either consonant or vowel).
- No separate character is available to specify half or compound characters in Braille. So, there is an identifier that is 4th dot to indicate that following character is half character.
- As Braille with 6 dots cells can represents only 64 characters, there are some Gujarati characters which are written in Braille as a combination of more than one character. For example, શ્ર in Gujarati is written in Braille as combination of 4th dot, શ, ર (3 characters).
- Braille word building also depends on the pronunciation of the words. It is spelled as it is pronounced.

So in Gujarati Braille recognition all the above-mentioned challenges are to be considered, otherwise the meaning may be changed.

4 Design Approach

The proposed algorithm is described, keeping in wits that the text file will be digital Braille text file which is in the form of Unicode, i.e., UTF-8 code. The Braille considered for designing algorithm is Standard Braille, i.e., American Grade 0. Following is the design approach for the recognition of Gujarati Braille Text:

Step 1: Read the file containing the Braille text.
Step 2: Extract each cell from the Braille text file.
Step 3: Convert extracted Braille cell to its Unicode.
Step 4: Store all Unicode of the text into an array.
Step 5: Traverse the whole array and repeat step 6–10.
Step 6: Retrieve element (Braille cell) from the array.
Step 7: Identify the code of element (Braille cell).
Step 8: If the code is of Gujarati letter (consonant and vowel) then
Map the code with the Mapping table and according to the code recognition is done for the cell.
Step 9: If the code is of digit identifier then
Retrieve next code from the array until space is encountered and map the code with the mapping table and according to it recognize the digits.
Step 10: If the code is of punctuation then
Match the code to the mapping table and recognize the Braille cell.

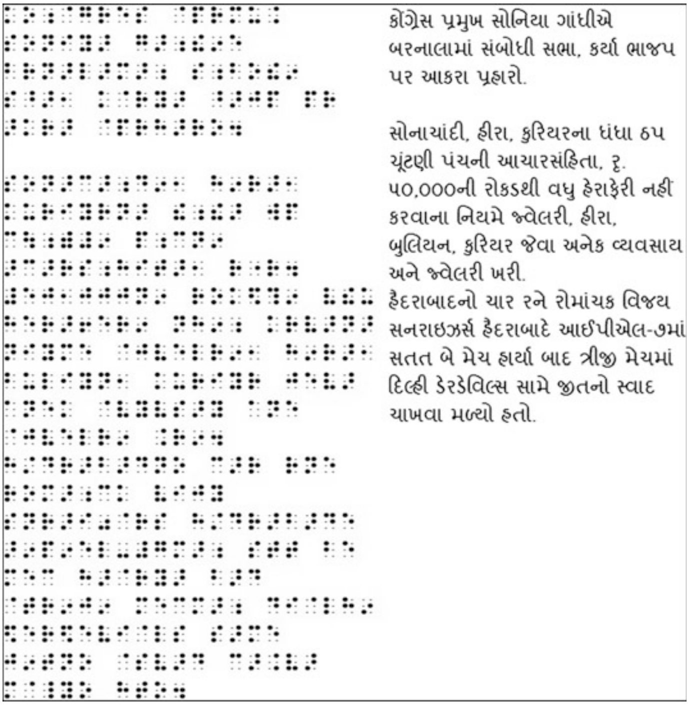
This algorithm will works on mapping technique.

If the proposed design is implemented then it will benefit the visually impaired people of the society. For example, if visually impaired person want to type some application, then that person has to take help of other sighted people and has to be dependent on the people who will type for them in Gujarati. But through proposed work, blind people will able to type their application in Braille and it will be recognized. So that sighted person will able to read.

5 Proposed Implementation and Result

Implementation will be done using MATLAB. Input will be the digital text file which contains text written in Braille and the text file will be in UTF-8 format. Braille cells will be extracted from the file and rules of Braille script will be applied to it. According to the rules and the recognition of the cell, character will be identified. Output will be the Gujarati text that is transliterated from the Braille text.

When the design approach described in topic 4 will be implemented, then it will be able to recognize all the Braille text in the documents present in the digital format, i.e., Unicode format and can be converted into the plain Gujarati text as shown in Fig. 7.



કોંગ્રેસ પ્રમુખ સોનિયા ગાંધીએ
બરનાલામાં સંબોધી સભા, કર્યા ભાજપ
પર આકરા પ્રહારો.

સોનાચાંદી, હીરા, કુરિયરના ધંધા ઠપ
ચૂંટણી પંચની આચારસંહિતા, રૂ.
૫૦,૦૦૦ની રોકડથી વધુ હેરાફેરી નહીં
કરવાના નિયમે જ્વેલરી, હીરા,
બુલિયન, કુરિયર જેવા અનેક વ્યવસાય
અને જ્વેલરી ખરી.
હૈદરાબાદનો ચાર રને રોમાંચક વિજય
સનરાઇઝર્સ હૈદરાબાદે આઈપીએલ-૭માં
સતત બે મેચ હાર્યા બાદ ત્રીજી મેચમાં
દિલ્હી ડેરડેવિલ્સ સામે જીતનો સ્વાદ
ચાખવા મળ્યો હતો.

Fig. 7 Proposed result

6 Conclusion

The paper describes that adequate research is carried out on the recognition of digital Braille text of various languages. But still Gujarati Braille Text recognition is un-touch important problem. In Braille languages like English, Arabic, and Hindi, the work is also found on Optical Character Recognition, but no work is found in Optical Gujarati Braille recognition. So, Gujarati Braille Text Recognition is the important problem to be solved. If the proposed algorithm is implemented then it will recognizes the Braille text and further it can be transliterated into Gujarati language. It will be benefitted to visually impaired people as they will also be able to communicate in written form and they do not have to rely on other people for their work.

References

1. Legge, G.E., Madison, C.M., Mansfield, J.S.: Measuring Braille reading speed with the MNREAD test. Vis. Impairment Res. J. **1**(3), 131–145 (1999)
2. Natural Language Processing. http://www.webopedia.com/TERM/N/NLP.html
3. Text Processing. http://www.macmillandictionary.com/dictionary/british/text-processing
4. Braille—The online encyclopedia. http://www.Wikipedia.com/Braille—Wikipediathefreeencyclopedia.htm
5. Chaudhary, A., Garg, P., Agarwal, A.: Using rotation method for removal of misalignment of scanned Braille pattern. In: Proceedings of the Second International Conference on Advances in Computing, Control and Communication, pp. 71–75 (2012)
6. Singh, M., Bhatia, P.: Automated conversion of English and Hindi text to Braille representation. Int. J. Comput. Appl. **4**(6), 25–29 (2009)
7. Blenkhorn, P.: A system for converting braille into print. IEEE Trans. Rehabil. Eng. **3**(2), 215–221 (1995)
8. Halder, S., Hasnat, A., Khatun, A., Bhattacharjee, D., Nasipuri, M.: Development of Bangla character recognition (BCR) system for generation of Bengali text from Braille notation. Int. J. Innovative Technol. Exploring Eng. **3**(1), 5–10 (2013)
9. Acharya-Multilingual Computing for Literacy and Education. www.acharya.gen.in/IntroductiontoBraille.htm
10. Rajarapollu, P., Kodolikar, S., Laghate, D., Khavle, A.: FPGA based Braille to text & speech for blind persons. Int. J. Sci. Eng. Res. **4**(4), 348–353 (2013)
11. Braille Cell Dimension, https://commons.wikimedia.org/wiki/File:Braille_code_dimensions.jpg
12. Shreekanth, T., Udayashankara, V.: An algorithmic approach for double sided Braille dot recognition using image processing techniques. Int. J. Image Process. Vis. Commun. **2**(4) (2014)
13. Bharati Braille Reference: Gujarati. http://www.acharya.gen.in:8080/cgi-bin/brcell_disp.pl?gujarati
14. Gardner–Salinas braille codes. https://en.wikipedia.org/wiki/Gardner%E2%80%93Salinas_braille_codes
15. Babu Suthar—Gujarati-English Learner's Dictionary. http://ccat.sas.upenn.edu/plc/gujarati/guj-engdictionary.pdf
16. Kayasth, M., Patel, B.: Offline typed Gujarati character recognition. Natl. J. Syst. Inf. Technol. **2**(1), 73–82 (2009)
17. Sojitra, B., Dhakad, V.: Neural network in character recognition of Gujarati script. J. Inf. Knowl. Res. Comput. Eng. **2**(2), 269–272 (2012)
18. Mousa, A., Hairy, H., Alomari, R., Alnemer, L.: Smart Braille system recognizer. Int. J. Comput. Sci. **10**(6), no. 1, 52–60 (2013)
19. Padmavathi, S., Reddy, S.S., Meenakshy, D.: Conversion of Braille to text in English, Hindi and Tamil languages. Int. J. Comput. Sci. Eng. Appl. **3**(3), 19–32 (2013)
20. Authman, Z., Jebr, Z.: Arabic Braille scripts recognition and translation using image processing techniques. J. Coll. Educ. **2**(3), 18–26 (2012)
21. Wong, L., Abdulla, W., Hussmann, S.: A software algorithm prototype for optical recognition of embossed Braille. In: Proceedings of the 17th International Conference on Pattern Recognition, IEEE, vol. 2, pp. 586–589 (2004)
22. Mennens, J., Tichelen, L., Francois, G., Engelen, J.: Optical recognition of Braille writing using standard equipment. IEEE Trans. Rehabil. Eng. **2**(4), 207–212 (1994)
23. Al-Saleh, A., El-Zaart, A., Al-Salman, A.: Dot detection of Braille Images using a mixture of beta distribution. J. Comput. Sci. **7**(11), 1749–1759 (2011)
24. Al-Salman, S., Fathi, S.: Arabic Braille recognition and transcription into text and voice. In: 5th Cario International Biomedical Engineering Conference, pp. 227–231 (2010)

Development of Embedded Platform for Sanskrit Grammar-Based Document Summarization

D. Y. Sakhare, Raj Kumar and Sudiksha Janmeda

Abstract Automatic summarization is taking out essential data from a huge amount of information and leaving the details which are not crucial. The paper proposes to develop the embedded platform for the domain-specific article summarization. Purva mimansa principles from the traditional Sanskrit shastras are used for the extraction. The summary is generated by MATLAB and then serially transmitted through Arduino board and displayed on LCD.

Keywords Sanskrit Shastra · Automatic summarization · Purva Mimamsa · Extractive text summarization

1 Introduction

Summary is a precise representation of the important concepts of the input documents. It also can be defined as a brief and accurate representation of the important concepts of the source documents. Basically, summary can be of two types extractive and abstractive. In extractive summarization, sentences are extracted as summary based on the benchmark features. The abstractive methods require the ability to make new sentences, which has its indigenous benefits indeed on reducing redundancy and keeping a good compression rate [1]. The paper focuses on extractive text summarization. The proposed model uses the mimansa principles

D. Y.Sakhare (✉)
Dept of Electronics Engineering, MIT AOE, Alandi, Pune, India
e-mail: diptiysakhare@gmail.com

D. Y.Sakhare
BVDUCOE, Pune, Maharashtra, India

R. Kumar
DIAT, Khadakawasala, Pune, Maharashtra, India

S. Janmeda
MIT AOE, Alandi, Pune, Maharashtra, India

S. S. Agrawal et al. (eds.), *Speech and Language Processing for Human-Machine Communications*, Advances in Intelligent Systems and Computing 664,
https://doi.org/10.1007/978-981-10-6626-9_5

from traditional Sanskrit shastras for feature extraction. The developed system uses NN for feature selection as well as for sentence formation [2, 3].

2 Literature Survey

Begum [4] proposed an approach called trainable summarizer. The summarizer is developed using support vector machine which in turn is trained by feature score function. The support vector machine (SVM) has been trained by using all the features score function in order to construct a text summarizer model.

Prabhakar and Chandra [5] proposed pragmatic analysis-based summarizer which uses word sense disambiguation. The system is better in generating the high-quality summaries as compared to manual summarization.

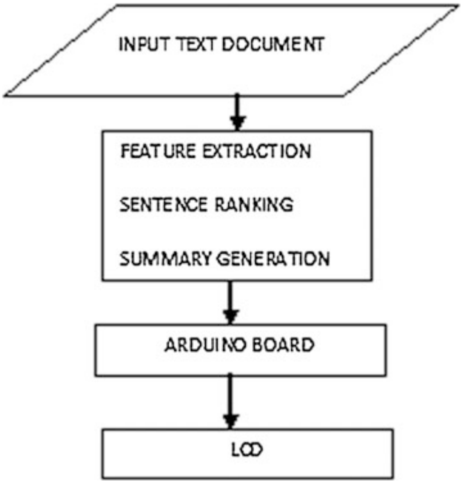
Kallimani and Reddy [6] developed a model for summarization for Kannada language. The summary facilitates the quick and accurate identification of the topic of the published document based on the statistical features. Various analyses of results were also discussed by comparing it with the English language. The results have clearly shown that the system (KanSum) is working efficiently on par with autosum concept. Its disadvantage is that it deals with only single document.

Salim [7] gives a survey on multi-document summarization approaches. They have elaborated 1. feature-based method, 2. cluster-based method, 3. graph-based method, and 4. Knowledge-based method. Based on the generic components, the paper has outlined a novel approach for news article summarization. The proposed method generates a good summary by applying various algorithms. It has got one more advantage that it generates summary of various types (like indicative, informative), so one can have desired summary.

3 Proposed Methodology

The primary intention of this work is to design and develop an embedded platform for efficient domain-specific summarization. Initially, a large document which is to be summarized is given to the system. This document goes through a number of preprocessing steps, e.g., segmentation of sentence, tokenization, removal of stop words, and word stemming. The features are extracted from the preprocessed document. The sentences having the higher feature score are selected for the summary. The summary extracted in MATLAB is transmitted to Arduino board serially. Arduino saves the file received and displays it on liquid crystal display (LCD). This is a prototype model which can be expanded for application-specific summary display, e.g., if the summary related to a particular sport is to be displayed in a stadium then this model can be used. The database used for the proposed method is real database which includes almost 150 news articles. Figure 1. Shows the flowchart of hardware and software assembly of this summarization system.

Fig. 1 Flowchart of the proposed method



4 Feature-Based Neural Network

The input text is transformed into the set of features, and this process is called as feature extraction. Traditional Sanskrit shastras have a great relevance in modern technology. Computing ideas are very well in Panini vyakaran of Sanskrit. Similarly, the text processing principles are explained in Sanskrit Mimamsa. [8]. Mimamsa principles are divided in two types: purva mimamsa and uttar mimamsa. Purva mimamsa principles (upkram upsamhar abhys apurvata phalam) are more related to abstractive summary while the uttar mimamsa principles (sthan, prakaran, samakhya) are more useful for extractive summary. The proposed approach uses the three principles from uttar mimamsa.

4.1 Prakaran

Prakarana is the one consistent meaning reached by a number of sentences meant to convey it, showing that all refers to it. The prakaran feature of an extracted document is calculated by comparing each sentence in the extracted document with the heading of that document. A check is performed against how many words are matched with the title. A sentence that contains the common words is ranked high.

4.2 Sthan

Sthana is the position reached in a discussion of a Prakarana. This feature is calculated as

$$\text{Sthan} = \frac{N - i + 1}{N} * \frac{n - m + 1}{n}, \tag{1}$$

N indicates the total number of paragraphs in the document,
n indicates the total number of sentences in a particular paragraphs.

4.3 Samakhya

Samakhya means ‘adding together.’ The related words appear together in the text many of the times, e.g., ‘heavy storm.’ Based on the mutual information and predefined length of window, the related words are extracted from the given text. The predefined length window rolled over the document from start to end. Then the co-occurring words are checked out. Based on the frequency of the selected words appearing together, the samakhya weight is calculated.

Then, for every concept extracted, the concept weight is computed based on their term frequency in every sentence and there after in every paragraph.

4.4 Summary Generation

To generate the summary, every sentence in a paragraph is represented as a row of a matrix. For this, a matrix of size *N***F* is generated where the number of sentences is given by *N* and *F* represents the features considered for the summary. As we are planning for domain-specific single document summarization application, the number of features chosen is less. Every element of the matrix is the feature score obtained for the corresponding sentence with the feature.

4.5 Training Phase

For learning purpose during the training phase, a feed-forward neural network is given the vector as an input. The target of the neural network signifies whether the sentence is useful for the summary or not [9]. Testing phase: the trained neural network is fed with the feature score of every sentence in the preprocessed document. The target of the neural network gives the sentences score for all the sentences in the document.

4.6 Sentence Ranking

Here, a weighted average formula is used to obtain the score of the sentences from the neural network. Get the sentence score from the neural networks. [10].

$$S = \lambda * S^F, \tag{2}$$

where $S \rightarrow$ Sentence score of the input sentence

$S^F \rightarrow$ Indicates the score of the sentence score given by the neural network.

$\lambda \rightarrow$ Indicates the constant of the weight.

Once the sentence score is obtained, all the sentences are ranked based upon their sentence score in descending order. As per the compression rate, the selected number of sentences is selected to be included in the summary. After that these sentences are arranged in the same order as they appear in the main text based on the unique ID.

4.7 Generation and Transmission

The output of the system is first displayed on MATLAB, in its command window. A text file is created by writing output data in it. The data of text file is transmitted into the Arduino board by MATLAB–Arduino interfacing.

5 Hardware

Here, we have interfaced Arduino with MATLAB and then output of the MATLAB is serially transmitted to LCD (interfaced with Arduino boards). The Arduino will receive data from the text file, and then the received serial data is saved in the board. Then the data is given to LCD for display purpose.

5.1 Arduino–MATLAB Interface

Firstly, the output data is stored in the variable $q5$, and then we write it in the text file (data saved in the form of arrays). After that, we will open our COM port by using the command (fopen). Then we write a suitable code (in ArduinoIDE), which will receive data from text file (via MATLAB code). After executing MATLAB code, the output will be stored in Arduino board.

5.2 Arduino–LCD Interface

Now, the serially transmitted data is stored in board and further given to LCD for display. We are using 20 * 4 LCD for display since it can show more number of characters on LCD which is necessary and sufficient for summary.

6 Results

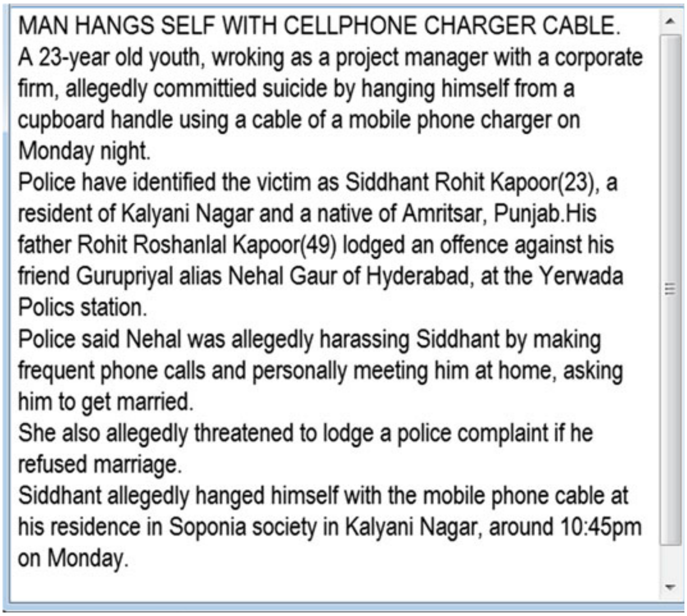
See Figs. 2, 3, and 4.

7 Evaluation Measure

F-measure, Recall, and Precision are used as evaluation measures of the system.

F-measure:

It is the measure of the results accuracy. It is calculated by given formula which is as follows:



MAN HANGS SELF WITH CELLPHONE CHARGER CABLE.
A 23-year old youth, wroking as a project manager with a corporate firm, allegedly committied suicide by hanging himself from a cupboard handle using a cable of a mobile phone charger on Monday night.
Police have identified the victim as Siddhant Rohit Kapoor(23), a resident of Kalyani Nagar and a native of Amritsar, Punjab.His father Rohit Roshanlal Kapoor(49) lodged an offence against his friend Gurupriyal alias Nehal Gaur of Hyderabad, at the Yerwada Polics station.
Police said Nehal was allegedly harassing Siddhant by making frequent phone calls and personally meeting him at home, asking him to get married.
She also allegedly threatened to lodge a police complaint if he refused marriage.
Siddhant allegedly hanged himself with the mobile phone cable at his residence in Soponia society in Kalyani Nagar, around 10:45pm on Monday.

Fig. 2 Input text document

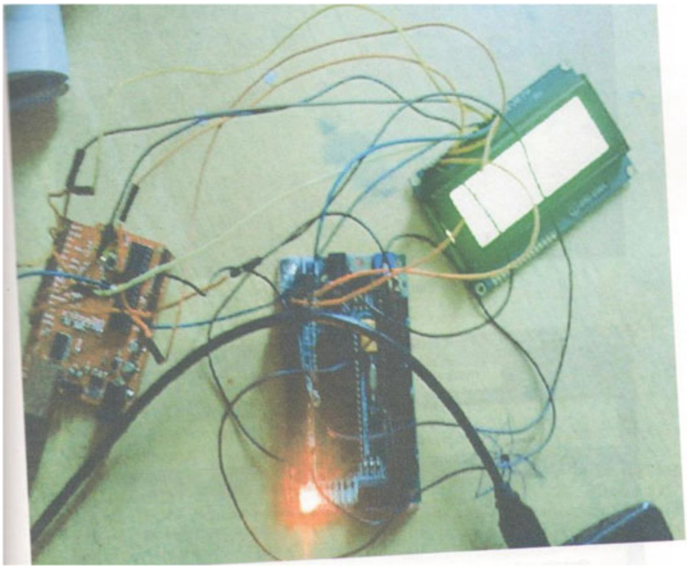


Fig. 3 Hardware view of the system_1

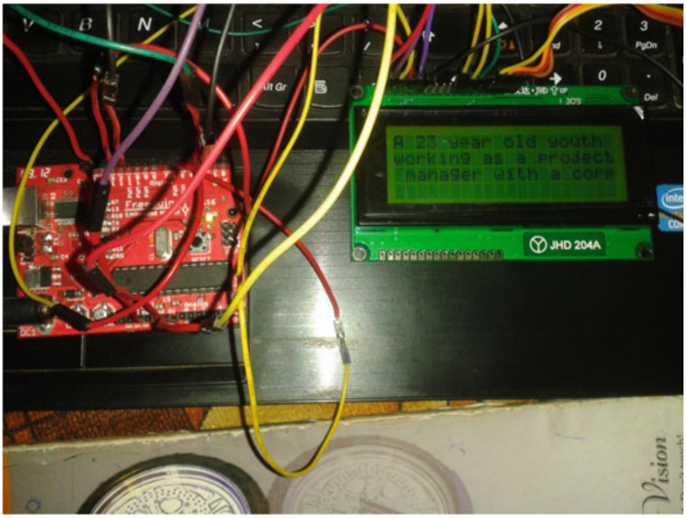


Fig. 4 Hardware view of the system_2

$$\mathrm{F-measure} = 2 \times \frac{\mathrm{Precision} + \mathrm{Recall}}{\mathrm{Precision} + \mathrm{Recall}}. \tag{3}$$

Precision:

It is the fraction of the retrieved sentences that are relevant. It is calculated by the equation given as follows:

$$\mathrm{Precision} = \frac{|\{\text{Retrieved sentences}\} \cap \{\text{Relevant sentences}\}|}{|\{\text{Retrieved sentences}\}|}. \tag{4}$$

Recall:

It is the fraction of the relevant sentences that are retrieved. It is calculated by the equation given as follows:

$$\mathrm{Recall} = \frac{|\{\text{Retrieved sentences}\} \cap \{\text{Relevant sentences}\}|}{|\{\text{Retrieved sentences}\}|}. \tag{5}$$

7.1 Calculations

Here, we have fixed relevant sentences (generated by the system) and variable retrieved information. These graphs will give précised information between the actual generated summary and the retrieved summary. In these plots, horizontal axis refers to the relevant information while vertical axis refers to either F-measure or Precision or Recall. Here, the compression ratio means the text of ten sentences, the important three or four or five lines are extracted out (Figs. 5, 6, 7, and 8; Tables 1, 2, and 3).

Conclusion A reliable summary should have F-measure values more than 0.5. As the summary generated by the proposed approach using purva mimansa has values greater than 0.5, which means the summary is reliable. In future, we will try to

Fig. 5 Recall for the compression ratio of 30%

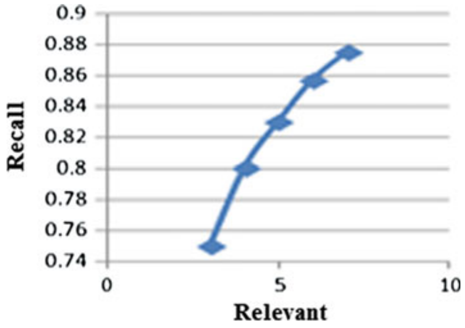


Fig. 6 Precision for the compression ratio of 30%

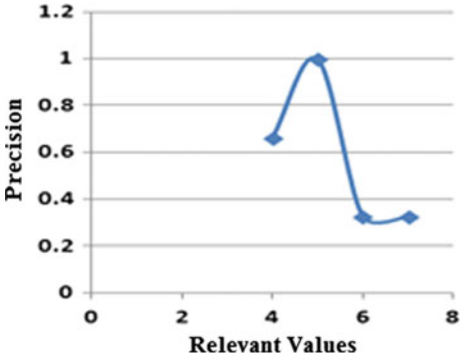


Fig. 7 Recall for the compression ratio of 40%

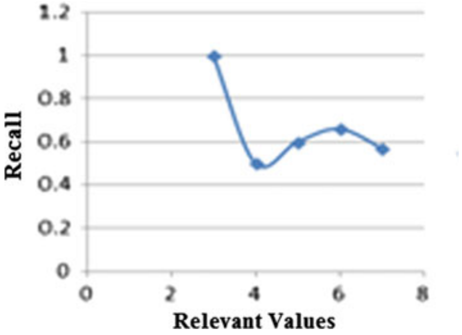


Fig. 8 Precision for the compression ratio of 30%

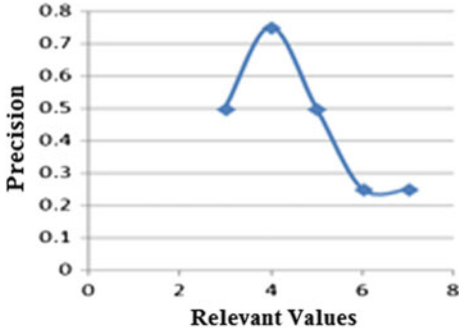


Table 1 Evaluation measure values for the compression ratio of 30%

Rel	Ret	Precision	Recall	F-measure
4	3	2/3 = 0.66	¾ = 0.75	0.7
5	3	3/3 = 1	4/5 = 0.8	0.72
6	3	1/3 = 0.33	5/6 = 0.83	0.73
7	3	1/3 = 0.33	6/7 = 0.857	0.745

Table 2 Evaluation measure values for the compression ratio of 40%

Rel.	Ret.	Precision	Recall	F-measure
3	4	2/4 = 0.5	3/3 = 1	0.66
4	4	¾ = 0.75	2/4 = 0.5	0.6
5	4	2/4 = 0.5	3/5 = 0.6	0.5
6	4	¼ = 0.25	2/6 = 0.66	0.33
7	4	¼ = 0.25	4/7 = 0.57	0.33

Table 3 Evaluation measure values for the compression ratio of 50%

Rel.	Ret.	Precision	Recall	F-measure
3	5	3/5 = 0.66	2/3 = 0.66	0.3
4	5	2/5 = 0.4	3/4 = 0.75	0.49
5	5	4/5 = 0.8	4/5 = 0.8	0.3
6	5	2/5 = 0.4	5/6 = 0.83	0.32
7	5	1/5 = 0.2	1/7 = 0.14	0.33

develop the summarization system using all Mimamsa principles. The applications which have to summarize short documents (typically 10–15 sentences) and where time and space are of major constraints can use the proposed model.

References

1. Foong, O.M., Oxley, A., Sulaiman, S.: Challenges and trends of automatic text summarization, Int. J Inf. Telecommun. Technol. **1**(1), 34–39 (2010)
2. Baxendale, P.B.: Machine-made index for technical literature: an experiment. IBM J. Res. Dev **2**(4), 354361 (1953)
3. Luhn, H.P.: Automatic creation of literature abstracts. IBM J. Res. Dev. 2(2), 159–165 (1958)
4. Begum, N., Fattah, M.A., Ren, F.: Automatic text summarization using support vector machine. Int. J. Innovative Comput. **5**, 1987–1996 (2009)
5. Prabhakar, M., Chandra, N.: Text summarization based on Pragmatic analysis. Int. J. Sci. Res. Publ. **2**, 34 (2012)
6. Kallimani, J.S.: Experiments with Ontology-based, customized, extractive text summary and word scoring. Cybern. Inf. Technol. **12**, 41 (2012)
7. Salim, N.: Automatic multi document summarization approaches. J. Comput. Sci. 8 **2**(1), 133–140 (2012)
8. Thibaut Phil, G.: The Arthsamgraha an elementary treatise on mimamsa. Banaras printing press (1882)
9. Kaikhah, K.: Automatic text summarization with neural networks. In: Second IEEE International Conference on Intelligent Systems (2004)
10. Sakhare, D.Y., Kumar, D.R.: Syntactic and sentence feature based hybrid approach for text summarization. I.J. Inf. Technol. Comput. Sci. 38–46 (2014)

Approach for Information Retrieval by Using Self-Organizing Map and Crisp Set

Mukul Aggarwal and Amod Kumar Tiwari

Abstract Nowadays, mostly users want to search anything they have done by Web search. For finding the concerned information with the reduced retrieval time they go through the search engine. Finding the relevant information with the help of mapped area affects the supervised and unsupervised learning method and works on the designing part of the information retrieval by using SOM and crisp set (CrispSOM), as well as reduces the retrieval time and collects the scattered information using self-organizing map and crisp set. The innovative idea of this paper is to evaluate the application of self-organizing maps (SOM) with the help of crisp value for finding the relevant information in lesser time as compared to other information retrieval system (IRS).

Keywords Database searching · Information retrieval · Self-organizing map · Crisp set

1 Introduction

Information word self-explanatory includes data, speech, image (2-D or 3-D), audio, and video. The main job of information retrieval is finding the text documents and the retrieval of other media things. In other thing is also important that representation, storage and organization of information and access all includes a huge variety of information-related job.

The SOM is a general tool for managing the unsupervised data into supervised data. The main purpose is to collect the same type of information or having this information which has same properties collected in one database (means cluster

M. Aggarwal (✉)
Department of Information Technology, KIET Group of Institutions, Ghaziabad, India
e-mail: mukul.aggarwal@kiet.edu; mukul.digital@gmail.com

A. K. Tiwari
Bhaba Institute of Technology, Kanpur, India
e-mail: amod@iitk.ac.in; amodtiwari@gmail.com

S. S. Agrawal et al. (eds.), *Speech and Language Processing for Human-Machine Communications*, Advances in Intelligent Systems and Computing 664,
https://doi.org/10.1007/978-981-10-6626-9_6

data sample). This regard problem rises on 3-D map because map is generally 2-D as it is knowledgable for a user, machine sound and visualized on a machine that mapping the information with current scenario. The beauty of this paper is crisp with SOM. It is not necessary that the chosen member or element is always in the set, because the crisp value is taken only from a set not from universal set. Fuzzy value allows that type of elements to be partially in a set. Every element in a set has a membership value. This membership value can range from [0, 1]. 0 stands for non-availability of the element, and 1 stands for availability of element in a set. In this paper will improve the SOM algorithms with the help of crisp set for finding the information fast and more accurately.

2 Related Work

1. Information retrieval can be more advanced by the relevancy of the search results for understanding the intention of search and the context of terms as entered [1]. With the growth of digitization resources, both on and offline data on the Web, and increased variety in types of scattered collections, futuristic information systems will face growing difficulties in providing reliable, useful, and timely information. Time is a ubiquitous factor at many stages in the information-seeking process, with users having temporally relevant information needs. This issue aims to explore opportunities and novel research on the intersection of time and information retrieval [2, 9]. Data fusion method measures by the effectiveness and similarity of search results through geometric framework [3].
2. Better knowledge of all type of patients location information search by pin codes or street wise is very problematic in health research for individual patients. Although, the emerging of location information makes it somehow very easier to other to determine the information of the patients. Other way to find the location information of individual is possible through aggregation [4].
3. Ontology-based information retrieval technique is mainly based on reasoning of the information users' query, by which the user's intention can be understood in better way and can return back the information to users. Then related information from the database is queried, which may be more than one. Relevance ranking of query to the information is necessary, and then a reasonable sequence of information is returned to the user, which allows users to find the information they needed quickly [5].
4. A model which works on the user's interest is named user interest model. Finding the information based on user personal information is called personalized information retrieval. In this model, all interest set of user as keyword list on client side can supply all personal information service for user with the help of communication media and merging with some architecture models. To find the most accurate information, have been done in a way that items are most relevant for their interests.

5. Other work on content-based finds the information, in which content filtering concept happens. And search based filtering content and specific features of the item the user is built on basic assumptions are required to be able to formulate a query to express the information needs and their interest.
6. By constructing a system for information retrieval with the help of SOM application. To evaluate the unsupervised and supervised data from set of all database which have exist any link with the premises value. So that SOM-based IRS provides the interfacing between two or more users that help them to find a basic view of related topic [6].
7. During 1991–1995, pioneer and Merk are the most productive researchers in information retrieval field. Their works mainly focus on feasibility, effectiveness, and usefulness of the SOM algorithms for ordering and maintaining the document. In this, model has to collect all scattered document and put in one database contains the collections of document. In this, model is something typical rather than other models because of involvement of all things in document form in one database like small collections of document about one hundred documents, 2-D maps of a hundred of neurons and vectors that represent documents, where the value represents itself for availability and non-availability of marked data in document collection. The main use of the SOM's standard is learning theory with Euclidean distance metric for finding the closest distance between two neurons.
8. Further will map the image by different mapping techniques and execute the action of mapped documents in set of neuron with respect to the user selection based on vector matrices accordingly. A search for keywords for support retrieval how many hits in each neuron for mapping the image, then after permit the selection of user action which help them for 2-D image retrieval [7].

3 Research Gap and Problem

SOMs and other various methods have been used for information retrieval [8]. In text retrieval, document retrieval, and other information including image in 2-D manner SOM method is useful for finding the information by clubbing some other methods like PicSOM, LabelSOM. The novel idea is retrieving the accurate information using by SOM and crisp set called CrispSOM [9, 10]. The main focus is on the development and evaluation of search engines. In nineteenth century, most usable devices such as laptop, tab, and other machines are used for retrieving the information by Web search at any time. Searching from Web is very daily routine work nowadays. So there are still till time taking processes and also find the concerning information is main issue. The various search engines are working on information retrieval by using different methods apart from SOM. SOM is basically artificial neural network model for unsupervised data. The huge amounts of data are available in Web in different formats like multimedia, image, audio, and video. Because of massive data available on internet, problematic issue is finding the relevant information.

The SOM methods are basically for clustering and converting the high 2-D data to a low-dimensional map data preserving the relations of the data set as distances on the mapped area. Converting the data from unsupervised to supervised data was the result of SOM only. SOM also gives visual representation of results with their map interface through simulation tools and the map view can be useful in visualization of classification results also.

The main highlighted problems in previous work are as follows:

- How SOM will search 3-D image from Web?
- How well SOM performs with other unsupervised method?
- Is it possible to develop the SOM method with crisp value for getting the better result in information retrievals?
- Is the generated map view beneficial for information retrieval system?
- How to improve or modify the artificial neural network model for retrieving the relevant information?

By merging the application of SOM with crisp sets in the above-specified questions in information retrieval. There are so many things which seem to be full of interesting questions and possibilities. The focus of this paper to generate output in form of results where the neurons are taken as input value regarding the SOM performance in information retrieval and provide best result against the other various used machine learning methods in image classification.

4 Impact on Academics/Industry

Main impact on academics for developing the search engines for finding relevant information according to user’s need as well as in industry. The developed search engines are available and freely usable to public implementing in the World Wide Web. The most important role of information retrieval is to automatically get the relevant information from a collection of huge data set based upon a search query generated by user. The proposed task would be helpful for generating the results about the CrispSOM performance for retrieving the information. There are so many different methods of the finding information process, depending on the user’s interest:

- For automatic application—Get the information automatically, just place a query.
- For Librarian—Librarian can maintain and organize their books and articles in better way and also helpful for indexing the information.
- For Cognitive scientist—Mainly used in biometric techniques with recognition of different things like iris, hand, thumb recognition.

5 Methodology

- Method for proposed work by self-organizing map and crisp set. In this firstly information may or may not be in scattered format. If information is in unscattered format, then collect all the information in data warehouse, and on the other hand, scattered information should be present in same data warehouse.
- By applying SOM technique for retrieval the information, data have both supervised and unsupervised data, so SOM converts unsupervised data to supervised data. Hence, we have all data in supervised manner in data warehouse. This data warehouse is called data mapped area. (Type of data mart in which data contain only supervised data).
- After applying SOM, we will use crisp set to sum up crisp with SOM, and the new technique CrispSOM would be evaluated. After that we can collect the filtered information.
- In filtered information, all types of supervised information will be present in the form text document, 2-D image, 3-D image, multimedia file, etc.
- The approach for information retrieval system by using novel method, the CrispSOM, can be divided into two categories.

  (i) Create the user interface environment which is helpful to make search engine available and freely usable to public implementing in the World Wide Web.

  (ii) Functional part in search engine has been implanted like image retrieval with file extensions image.cgi which holds all requests and responses from the user side. It also includes saving and updating the information from previous queries and executing that part as required for completing the request.

6 Conclusion

This paper focused on the efficient new techniques by sum up of self-organizing map and crisp set. The main work will be focusing on improving the SOM algorithm in searching and finding the most accurate relevant information with reduced retrieval time that will be used for supplying the query with reasons of the search results. The advantages of this approach for academics and industry people are maintaining, handling, and assessing the information from Web. The user want to give their valuable time during their study so could retrieve the information using search queries. Future scope, the information system is expected to provide some 3-D images clubbed with some other information. 3-D image and text will be used by the retrieved image with data in same or different high-dimensional clustering. This is very excited research area nowadays where different types of information are retrieved with different methods.

References

1. Remi, S., Varghse, S.C.: Domain ontology driven fuzzy semantic information retrieval. Procedia Comput. Sci. **46**, 676–681 (2015)
2. Derczynski, L., Strötgen, J., Campos, R., Alonso, O.: Time and information retrieval: introduction to the special. Inf. Proces. Manage. 1–5 (2015)
3. Wu, S., Crestani, F.: A geometric framework for data fusion in information retrieval. Inf. Syst. **50**, 20–35 (2015)
4. Dankar, F.K., Emam, K.El., Matwin, S.: Efficient private information retrieval for geographical aggregation. Procedia Comput. Sci. **37**, 497–502 (2014)
5. Wulamu, A., Zhou, Y., Zhang, D., Li, H., Rui, H.: The research and application of ontology-based information retrieval. In: IEEE 9th conference on industrial electronics and applications (ICIEA), pp. 1980–1984 (2014)
6. Gong, S.: The personalized information retrieval model based on user interest. Phys. Procedia **24(**B), 817–821 (2012)
7. Fernandes, R., Pinheiro, B.F.: Self-organizing maps applied to information retrieval of dissertations and theses from BDTD–UFPE. In: Eleventh Brazilian symposium on neural networks, pp. 31–36 (2010)
8. Saarikoski, J., Laurikkala, J., Järvelin, K., Juhola, M.: A study on the use of self organizing maps in information retrieval. J. Doc. 1–41 (2009)
9. Rich, E., Knight, K.: Artificial intelligence. McGraw-Hill, New York (2009)
10. Han, J., Kambler, M.: Data mining: concept and techniques. Morgan Kaufmann publication (2000)

An Automatic Spontaneous Speech Recognition System for Punjabi Language

Yogesh Kumar and Navdeep Singh

Abstract Punjabi is a very tonal language, making employ of a range of tones to distinguish words that would otherwise be alike. Three main tones can be recognized: high-rising-falling, mid-rising-falling, and low rising. Some work has been done in the field of isolated word, connected word, and continuous speech recognition system for Punjabi language. Spontaneous speech recognition is one area where no work has been done so far for Punjabi language. Spontaneous speech and speech from written language are exceptionally dissimilar both acoustically and linguistically. Spontaneous speech contains crammed silence, preservation, faltering, duplications, incomplete vocabulary, and stuttering. In this paper, an effort has been made to build an automatic spontaneous speech recognizer to recognize Punjabi live speech by using speech recognition model using sphinx toolkit.

Keywords Acoustic model · Feature vector · Sphinx · Decoder · Phones
Transcript · Filler dictionary

1 Introduction

Dealing with unplanned speech [1] is one of the numerous challenges that Automatic Speech Recognition (ASR) systems for Punjabi language have to compact with. The primary indications describing spontaneous speech are hesitating like packed pause, repetition, repair and false start and many learning have paying attention on the recognition and improvement of these hesitations [2]. Therefore, identification of spontaneous speech will need a standard move from speech to accepting where original messages of the speaker are removed, as a

Y. Kumar (✉)
Department of Computer Engineering, Punjabi University Patiala, Patiala, Punjab, India
e-mail: Yogesh.arora10744@gmail.com

N. Singh
Department of Computer Science, Mata-Gujri College, Sri Fatehgarh Sahib, Punjab, India
e-mail: navdeep_jaggi@yahoo.com

S. S. Agrawal et al. (eds.), *Speech and Language Processing for Human-Machine Communications*, Advances in Intelligent Systems and Computing 664,
https://doi.org/10.1007/978-981-10-6626-9_7

substitute of transcribing every vocal words. Spontaneous speech, as evaluated to designed speech, is a more natural way in which people communicate with each other. However, the recognition of spontaneous speech is facing numerous challenging by the rigorous articulation alternatives and changeable silence gaps or amusement in between words. Presently, a variety of novel applications of LVCSR (large vocabulary continuous speech recognition) systems, such as automatic closed captioning, making minutes of meetings, conferences, and summarizing and indexing of speech documents for information retrieval, are dynamically being explored.

2 Automatic Spontaneous Speech Recognition System for Punjabi

Speech recognition [3] is a complicated task and states of the ability recognition systems are very complex. Automatic spontaneous speech has many prospective purposes including rule and organize, transcription of confirmed dialogue, live speech, and interactive vocal conversations (Fig. 1).

The primary phase [4] of speech identification is to reduce the speech signals into flows of acoustic feature vectors, called as *observations*. The key chore [5] of the speech system is to obtain an audio signal as input and fabricate a sequence of words as output. The acoustic model begins a mapping among phonemes and their potential acoustic demonstrations, i.e., the phones. The prior probability is computed using the language model. Usually trigram or even 4-g supported language models are utilized in recent speech systems. The decoding method [6] in a speech recognizer's procedure is to discover a string of words whose consequent acoustic and language models finest equivalent the input feature vector string. For that reason, the procedure of such a decoding process with trained audio and language models is often submitted to as a explore method.

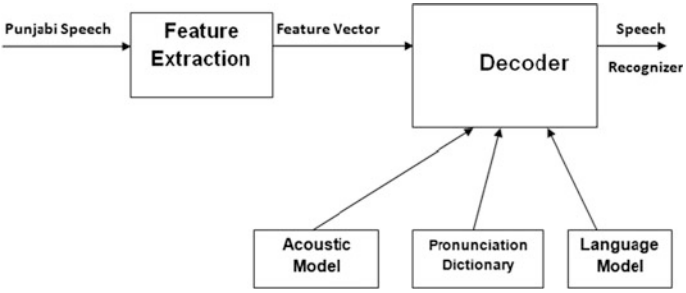


Fig. 1 Automatic speech recognition system for Punjabi speech

3 Building an Acoustic Model for Spontaneous Punjabi Speech

In order to build an acoustic model for spontaneous Punjabi speech, it is required to train the system with word level. But the single word wav file has small in size and silence gap is more therefore even for training single word, we need sentences. For this purpose, we trained the Punjabi spontaneous speech system with multiple words and sentences with variable silence gap.

A. Steps for training the acoustic model for Punjabi corpus

To train the system for Punjabi Language, we need following configuration files:

1. **Dic (Independent words are store in it)**:

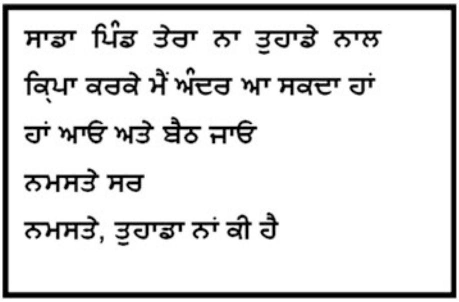
The main purpose of the dictionary file is to map Punjabi stored words with the every recorded Punjabi sound unit associated with each sounds. Two types of the dictionaries are present, first type is used in which reasonable words in the language are planned progressions of sound units, and second type of dictionary in which non-vocalizations sounds are planned to corresponding non-vocalizations or speech-like sound units is also created. The training data which we are giving as input to our system are shown in the given figure [7, 8] (Fig. 2).

The dictionary file (Punjabi.dic file) will look like as shown in Fig. 3:

2. **Filler and noise**: It is also type of dictionary in which rejected noise is stored [2]. For example:

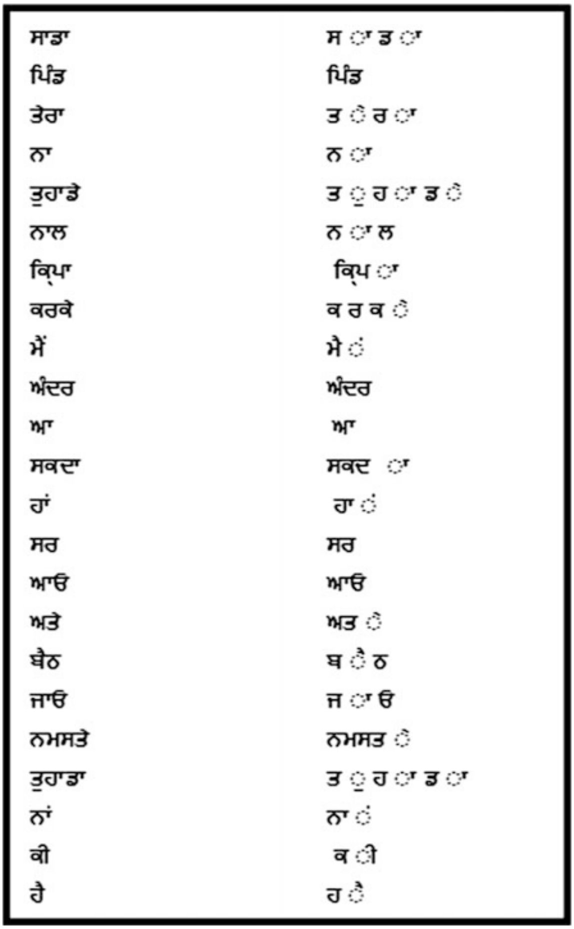
```
<s >     SIL
</s >    SIL
<sil >   SIL
```

3. **Phone**: Phone file [9] is a record of individual sound unit that needs to make a word. The various phone files are shown in the Table 1.



ਸਾਡਾ ਪਿੰਡ ਤੇਰਾ ਨਾ ਤੁਹਾਡੇ ਨਾਲ
ਕ੍ਰਿਪਾ ਕਰਕੇ ਮੈਂ ਅੰਦਰ ਆ ਸਕਦਾ ਹਾਂ
ਹਾਂ ਆਓ ਅਤੇ ਬੈਠ ਜਾਓ
ਨਮਸਤੇ ਸਰ
ਨਮਸਤੇ, ਤੁਹਾਡਾ ਨਾਂ ਕੀ ਹੈ

Fig. 2 Training data of Punjabi language



ਸਾਡਾ	ਸ ◌ਾ ਡ ◌ਾ
ਪਿੰਡ	ਪਿੰਡ
ਤੇਰਾ	ਤ ◌ੇ ਰ ◌ਾ
ਨਾ	ਨ ◌ਾ
ਤੁਹਾਡੇ	ਤ ◌ੁ ਹ ◌ਾ ਡ ◌ੇ
ਨਾਲ	ਨ ◌ਾ ਲ
ਕ੍ਰਿਪਾ	ਕ੍ਰਿਪ ◌ਾ
ਕਰਕੇ	ਕ ਰ ਕ ◌ੇ
ਮੈਂ	ਮੈ ◌ਂ
ਅੰਦਰ	ਅੰਦਰ
ਆ	ਆ
ਸਕਦਾ	ਸਕਦ ◌ਾ
ਹਾਂ	ਹਾ ◌ਂ
ਸਰ	ਸਰ
ਆਓ	ਆਓ
ਅਤੇ	ਅਤ ◌ੇ
ਬੈਠ	ਬ ◌ੈ ਠ
ਜਾਓ	ਜ ◌ਾ ਓ
ਨਮਸਤੇ	ਨਮਸਤ ◌ੇ
ਤੁਹਾਡਾ	ਤ ◌ੁ ਹ ◌ਾ ਡ ◌ਾ
ਨਾਂ	ਨਾ ◌ਂ
ਕੀ	ਕ ◌ੀ
ਹੈ	ਹ ◌ੈ

Fig. 3 Dictionary files of Punjabi corpus

Table 1 Phone files of Punjabi language

ਸ	ਡ	ਤ	ਰ	ਨ	ਹ	ਲ	ਕ
ਪ	ਜ	ਦ	ਠ	ਅੰ	ਆ	ਪਿੰ	ਕ੍ਰਿ
ਮੇ	ਹਾ	ਓ	ਅ	ਬ	ਮ	ਨਾ	◌ੁ
◌ੇ	◌ਾ	◌ਂ	◌ੈ	◌ੀ			

4. **Transcript** (path of wav files) **and Fields** (conversation of wav File):

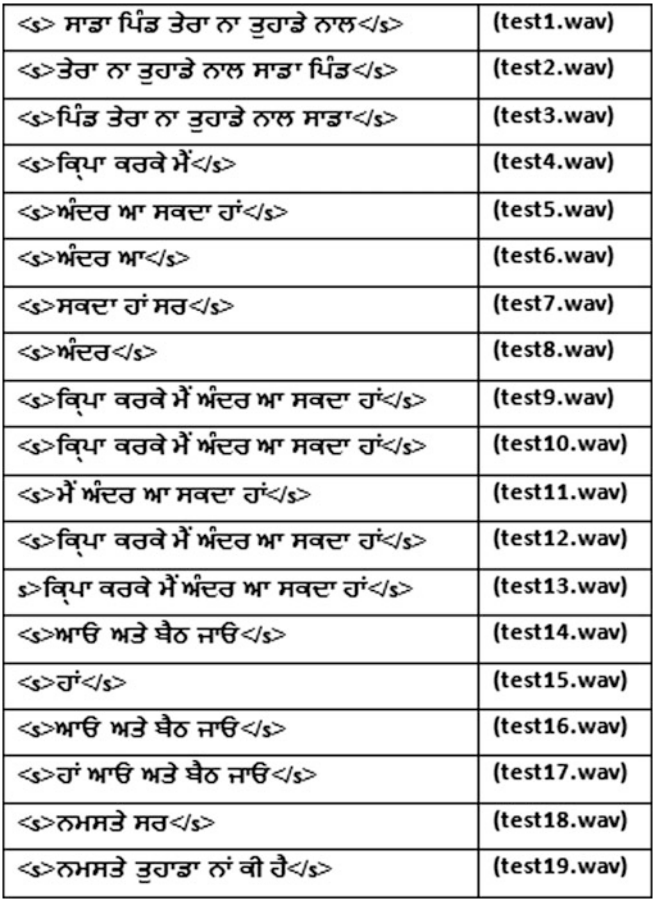
Transcription file is a listing the dictation for each acoustic file. For example, in our Punjabi corpus, the Table 2 shows the transcription file for test audio:

It is essential that each line of Punjabi text begins by <s> and finishes by </s> followed by id in parentheses. Also note that parenthesis includes only the file, exclusive of speaker_n directory. It is vital to have correct match among fields file and the transcription file.

We have two kinds of transcript and field files:

- **For training purpose** (Punjabi_parpare.trans and Punjabi_parpare.fileds)
- **For testing purpose** (Punjabi_check.trans and Punjabi_check.fileds)

Table 2 Transcript file



<s> ਸਾਡਾ ਪਿੰਡ ਤੇਰਾ ਨਾ ਤੁਹਾਡੇ ਨਾਲ</s>	(test1.wav)
<s>ਤੇਰਾ ਨਾ ਤੁਹਾਡੇ ਨਾਲ ਸਾਡਾ ਪਿੰਡ</s>	(test2.wav)
<s>ਪਿੰਡ ਤੇਰਾ ਨਾ ਤੁਹਾਡੇ ਨਾਲ ਸਾਡਾ</s>	(test3.wav)
<s>ਕ੍ਰਿਪਾ ਕਰਕੇ ਮੈਂ</s>	(test4.wav)
<s>ਅੰਦਰ ਆ ਸਕਦਾ ਹਾਂ</s>	(test5.wav)
<s>ਅੰਦਰ ਆ</s>	(test6.wav)
<s>ਸਕਦਾ ਹਾਂ ਸਰ</s>	(test7.wav)
<s>ਅੰਦਰ</s>	(test8.wav)
<s>ਕ੍ਰਿਪਾ ਕਰਕੇ ਮੈਂ ਅੰਦਰ ਆ ਸਕਦਾ ਹਾਂ</s>	(test9.wav)
<s>ਕ੍ਰਿਪਾ ਕਰਕੇ ਮੈਂ ਅੰਦਰ ਆ ਸਕਦਾ ਹਾਂ</s>	(test10.wav)
<s>ਮੈਂ ਅੰਦਰ ਆ ਸਕਦਾ ਹਾਂ</s>	(test11.wav)
<s>ਕ੍ਰਿਪਾ ਕਰਕੇ ਮੈਂ ਅੰਦਰ ਆ ਸਕਦਾ ਹਾਂ</s>	(test12.wav)
s>ਕ੍ਰਿਪਾ ਕਰਕੇ ਮੈਂ ਅੰਦਰ ਆ ਸਕਦਾ ਹਾਂ</s>	(test13.wav)
<s>ਆਓ ਅਤੇ ਬੈਠ ਜਾਓ</s>	(test14.wav)
<s>ਹਾਂ</s>	(test15.wav)
<s>ਆਓ ਅਤੇ ਬੈਠ ਜਾਓ</s>	(test16.wav)
<s>ਹਾਂ ਆਓ ਅਤੇ ਬੈਠ ਜਾਓ</s>	(test17.wav)
<s>ਨਮਸਤੇ ਸਰ</s>	(test18.wav)
<s>ਨਮਸਤੇ ਤੁਹਾਡਾ ਨਾਂ ਕੀ ਹੈ</s>	(test19.wav)

Training files are used to create feature vector which will be used later for recognition. Testing files are used by decoder to check the recognition. **Sphinx_train.test file**: This is the configuration file where configuring the path for all required files (for field, transcript, etc.).

4 Steps of Creating the Language Model for Punjabi Corpus

Language model is used for decoding purpose. The language model gives framework to differentiate between words and expression that sounds alike. There are two forms of language models [10] that illustrate language—grammars and statistical language models [11, 12]. Grammars portray very simple forms of languages for grasp and organize, and they are usually written manually or produced mechanically with plain code [13, 14]. Steps for creating language model are:

Step1: During compilation, first we input given text file as shown in Fig. 4.
Step2: Execute cmu command and create vocab file (Fig. 5).
Step3: Finally, language model is created with extension lm.DMP, which is used for training purpose. While training it use decoder to test the training and generate log files of decoding.

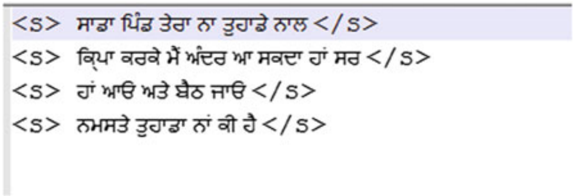
Figure 6 clearly shows that while compiling the Punjabi acoustic model for spontaneous speech, out of 128 lines and 390 words, only 2 lines and 1 word are failed. So the sentence error rate is 1.6% and word error rate is 0.5%.

5 Graphical User Interface for Automatic Spontaneous Speech System for Punjabi Language

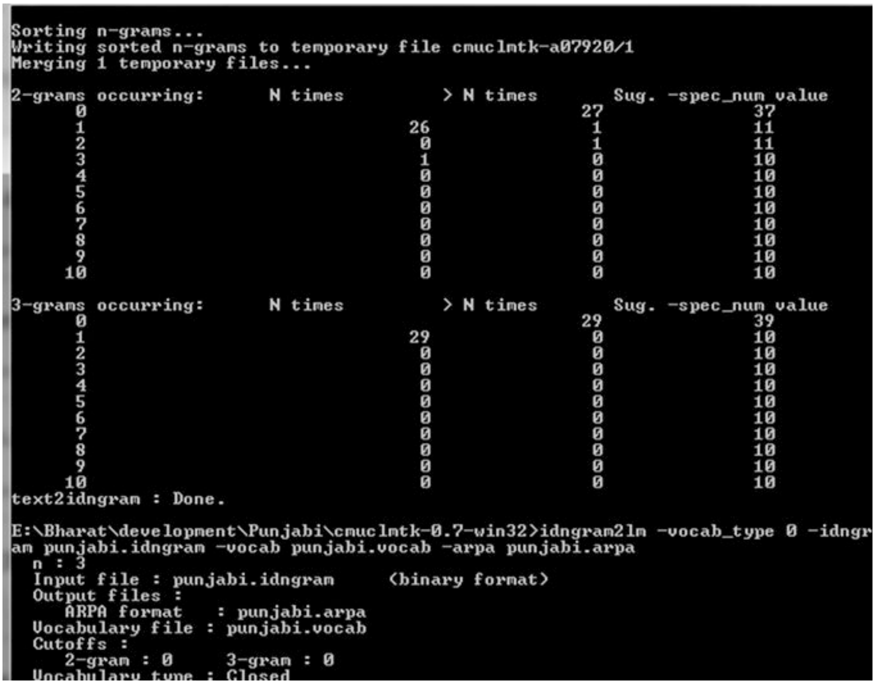
Language model and training data are both compiled in final jar file which is used for recognition. For live testing of speech, we have created the java based GUI for spontaneous Punjabi speech (Fig. 7).

It has an option of live speech test and speech recognition for already recorded wav files.

Fig. 4 Input Punjabi text file



```
<s>  ਸਾਡਾ ਪਿੰਡ ਤੇਰਾ ਨਾ ਤੁਹਾਡੇ ਨਾਲ </s>
<s>  ਕ੍ਰਿਪਾ ਕਰਕੇ ਮੈਂ ਅੰਦਰ ਆ ਸਕਦਾ ਹਾਂ ਸਰ </s>
<s>  ਹਾਂ ਆਉ ਅਤੇ ਬੈਠ ਜਾਉ </s>
<s>  ਨਮਸਤੇ ਤੁਹਾਡਾ ਨਾਂ ਕੀ ਹੈ </s>
```



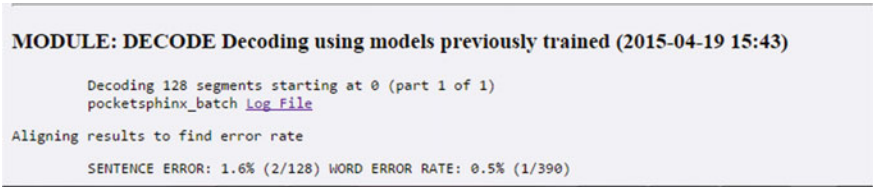
```
Sorting n-grams...
Writing sorted n-grams to temporary file cmuclmtk-a07920/1
Merging 1 temporary files...

2-grams occurring:      N times          > N times       Sug. -spec_num value
      0                                          27                 37
      1                           26              1                 11
      2                            0              1                 11
      3                            1              0                 10
      4                            0              0                 10
      5                            0              0                 10
      6                            0              0                 10
      7                            0              0                 10
      8                            0              0                 10
      9                            0              0                 10
     10                            0              0                 10

3-grams occurring:      N times          > N times       Sug. -spec_num value
      0                                          29                 39
      1                           29              0                 10
      2                            0              0                 10
      3                            0              0                 10
      4                            0              0                 10
      5                            0              0                 10
      6                            0              0                 10
      7                            0              0                 10
      8                            0              0                 10
      9                            0              0                 10
     10                            0              0                 10
text2idngram : Done.

E:\Bharat\development\Punjabi\cmuclmtk-0.7-win32>idngram2lm -vocab_type 0 -idngr
am punjabi.idngram -vocab punjabi.vocab -arpa punjabi.arpa
  n : 3
  Input file : punjabi.idngram     (binary format)
  Output files :
     ARPA format    : punjabi.arpa
  Vocabulary file : punjabi.vocab
  Cutoffs :
     2-gram : 0      3-gram : 0
  Vocabulary type : Closed
```

Fig. 5 1-, 2-, and 3-g after compiling vocab file



MODULE: DECODE Decoding using models previously trained (2015-04-19 15:43)

```
        Decoding 128 segments starting at 0 (part 1 of 1)
        pocketsphinx_batch Log File

Aligning results to find error rate

        SENTENCE ERROR: 1.6% (2/128) WORD ERROR RATE: 0.5% (1/390)
```

Fig. 6 Output of the decoder for Punjabi corpus

Figure 8 clearly shows that the output of the live speech testing for spontaneous Punjabi speech.

6 Performance Evaluation

The performance of the research work is evaluated by comparing it with previous work done for small vocabulary system [5]. In the previous research, the total numbers of sentences were taken 7 and words were 42 of Punjabi language [15, 16, 17]. The present work has total 128 sentences and 390 words. Table 3 shows the comparison between the previous and present work on the basis of sentences error and word error rate.

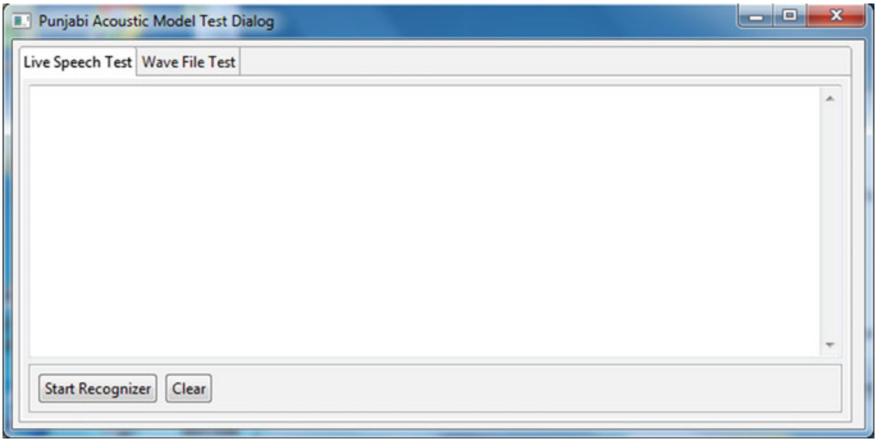
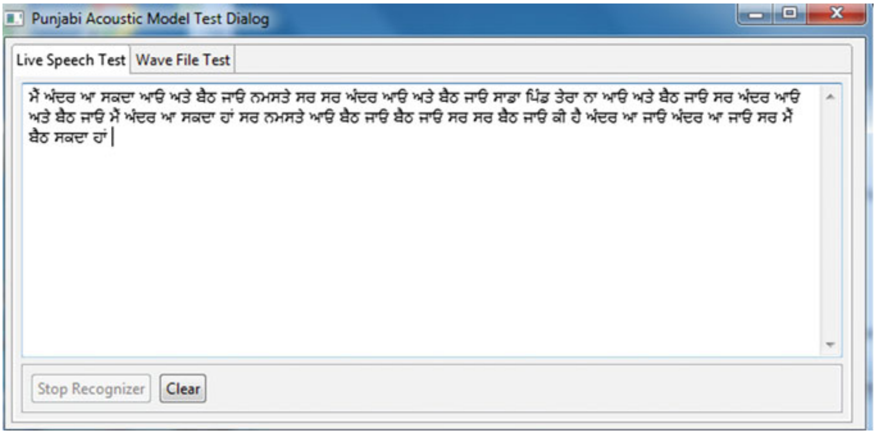


Fig. 7 GUI for spontaneous Punjabi speech recognition



ਮੈਂ ਅੰਦਰ ਆ ਸਕਦਾ ਆਓ ਅਤੇ ਬੈਠ ਜਾਓ ਨਮਸਤੇ ਸਰ ਸਰ ਅੰਦਰ ਆਓ ਅਤੇ ਬੈਠ ਜਾਓ ਸਾਡਾ ਪਿੰਡ ਤੇਰਾ ਨਾ ਆਓ ਅਤੇ ਬੈਠ ਜਾਓ ਸਰ ਅੰਦਰ ਆਓ
ਅਤੇ ਬੈਠ ਜਾਓ ਮੈਂ ਅੰਦਰ ਆ ਸਕਦਾ ਹਾਂ ਸਰ ਨਮਸਤੇ ਆਓ ਬੈਠ ਜਾਓ ਬੈਠ ਜਾਓ ਸਰ ਸਰ ਬੈਠ ਜਾਓ ਕੀ ਹੈ ਅੰਦਰ ਆ ਜਾਓ ਅੰਦਰ ਆ ਜਾਓ ਸਰ ਮੈਂ
ਬੈਠ ਸਕਦਾ ਹਾਂ

Fig. 8 Output of the Punjabi spontaneous speech recognition model

Table 3 Result comparison

Previous work		Present work	
Total number of sentences 7	Total number of words 42	Total number of sentences 128	Total number of words 390
Sentence error rate 28.6%	Word error rate 4.8%	Sentence error rate 1.6%	Word error rate 0.5%

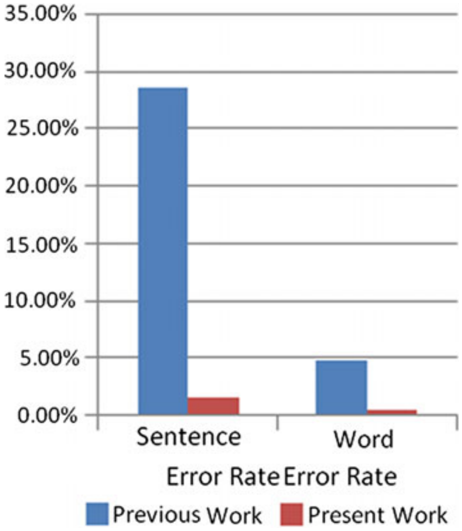


Fig. 9 Performance comparison

Graphical analysis shown in Fig. 9 represents drastic reduction in the word and sentence error rate with increase in vocabulary size in the previous and present work.

7 Conclusion and Future Work

In this paper, an effort has been made to develop an automatic spontaneous speech recognition system for Punjabi corpus using sphinx toolkit. The accomplishment of spontaneous speech detection system has considerably improved in provisions of sentence along with word error rate. GUI has been created to test the live Punjabi speech using java framework. In future, system will be trained for large vocabulary so that recognition rate can be improved for voice input taken from the different person. The Language model will also be improved in future work for fast decoding and recognition.

References

1. Atassi, H., Smékal, Z.: Emotion recognition from spontaneous slavic speech. In: 3rd IEEE International Conference on Cognitive Info Communications, 2–5 December 2012
2. Furui, S.: Spontaneous speech recognition and summarization. In: Proceedings IEEE Workshop on Spontaneous Speech Processing and Recognition (2010)

3. Singh, P., Dutta, K.: Formant analysis of punjabi non-nasalized vowel phonemes. In: The International Conference on Computational Intelligence and Communication Systems, pp. 375–380, Proceedings IEEE (2011)
4. Dua, M., Aggarwal, R.K.: Punjabi automatic speech recognition using HTK. IJCSI Int. J. Comput. Sci. Issues **9**(4), No 1 (2012)
5. Kumar, Y., Singh, N.: A first step towards an automatic spontaneous speech recognition system for Punjabi language. Int. J. Stat. Reliab. Eng. **2**(1), 81–93 (2015)
6. http://research.microsoft.com/pubs/118769/Book-Chap-HuangDeng2010.pdf
7. www.shabdkosh.com/pa/…/corpus/corpus-meaning-in-Punjabi-English
8. https://corplinguistics.wordpress.com/tag/punjabi/
9. http://cmusphinx.sourceforge.net/
10. www.speech.cs.cmu.edu/sphinx/doc/Sphinx.html
11. Sixtus, A., Molau, S., Kanthak, S.: Spontaneous speech characterization and detection in large audio database. In: SPECOM'2009, St. Petersburg, 21–25 June 2009
12. Izzad, M., Jamil, N.: Speech/non-speech detection in malay language spontaneous speech. In: The proceedings IEEE 2013, pp 219–224 (2013)
13. Shih, P.O., Chen, B.W.: Enhanced lengthening cancellation using bidirectional pitch similarity alignment for spontaneous speech. In: The international Symposium on Chinese Spoken Language Processing Proceedings (2012)
14. Ghai, W., Singh, N.: Analysis of automatic speech recognition systems for Indo-Aryan languages: Punjabi a case study. Int. J. Soft Comput. Eng. (IJSCE), **2**(1) March 2012. ISSN: 2231–2307
15. Ghai, W., Singh, N.: Tri-phone based acoustic modeling on continuous speech recognition for Punjabi language. IJCA, **72** (2013)
16. Hu, X., Wu, Y.: Collecting sentences from web resources for constructing spontaneous Chinese language model. In: The International Symposium on Chinese Spoken Language Processing Proceedings (2012)
17. Akita, Y., Kawahara, T.: Statistical transformation of language and pronunciation models for spontaneous speech recognition. In: The IEEE Transactions on Audio, Speech, and Language Processing, **18**(6) (2010)

A System for the Conversion of Digital Gujarati Text-to-Speech for Visually Impaired People

Nikisha Jariwala and Bankim Patel

Abstract In the epoch of hi-tech development, study on Text-to-Speech conversion shows remarkable enhancement in last couple of decades. Visually impaired people are not able to read, so Text-to-Speech system acts as an aid for visually impaired people for reading by hearing the text. In this paper, we presented the development of computer-based Gujarati Text-to-Speech system that delivers text in Gujarati audio form. Arbitrary digital Gujarati text is considered as an input to the system; conversion is done with regard to the Akshara of Gujarati language, and sound is produced in the form of phoneme, diphone, or syllable as per the requirement. Single audio file is created of the text so that it can also be heard at later stage. The detailed algorithm along with the format of speech database is also presented in the paper. Proposed system is tested on the documents collected from online news Web site and it gives satisfactory result.

Keywords Text-to-Speech (TTS) · Speech synthesis · Natural language processing · Text processing

Ms. Nikisha B. Jariwala, Ph.D. Scholar & Asst. Professor of Smt. Tanuben & Dr. Manubhai Trivedi College of Information Science, affiliated to Veer Narmad South Gujarat University, Surat, Gujarat, India.
Dr. Bankim Patel, Director, Shrimad Rajchandra Institute of Management & Computer Application, Uka Tarsadia University, Maliba Campus, Gujarat, India.

N. Jariwala (✉)
Smt. Tanuben & Dr. Manubhai Trivedi College of Information Science,
Surat, Gujarat, India
e-mail: njariwala@acm.org

B. Patel
Shrimad Rajchandra Institute of Management & Computer Application,
Uka Tarsadia University, Surat, Gujarat, India
e-mail: bankim.patel@utu.ac.in

S. S. Agrawal et al. (eds.), *Speech and Language Processing for Human-Machine Communications*, Advances in Intelligent Systems and Computing 664,
https://doi.org/10.1007/978-981-10-6626-9_8

1 Introduction

Language is used by the people as the means of communication. Speech [1] is the ability to express the thoughts and emotions. Speech is a form of communication that is based on combination of natural sound in the form of units. When these units are kept together, they may form a word or a sentence. The smallest sound unit in the sound wave that has definite shape is known as phone [2]. A group of phones that compose perceptually distinctive units is called phoneme. In a verbal sequence, a pair of phonetic sounds kept adjacent to each other is called diphone [3]. Diphthong is a similar language element as diphone. A combined sound that contains two or more vowel components is known as diphthong. Two separate sounds, consisting of either vowels or consonants, placed next to each other are referred as diphone. Syllable [4] is one or more letters representing a unit of spoken language consisting of a single uninterrupted sound. It is made up of either a single vowel sound or a combination of vowel and consonant. A standalone syllable, i.e., a single syllable is called a monosyllable, whereas the combination of two or more syllables in a word is called polysyllable.

Text [5] is a human-readable character sequence that can be encoded in the computer-understandable format such as ASCII and Unicode—a standard character set encoding; used worldwide to develop e-content, i.e., UTF-8 format. Text processing refers to the manipulation of the text, i.e., more precise transformation of text from one format to another.

Text-to-Speech (TTS) is a system that converts linguistic text into spoken voice output. It is also referred as speech synthesis that artificially produces human speech. So the objective of TTS tool [6] is to automatically convert written text to corresponding speech, and it can be used for various purposes such as to hear content by visually impaired people, public announcement at the railway stations or at airports, and in telephone services provided by banks and call centers to retrieve information. To make TTS system more understandable and natural, Rhythm [2] is a significant factor. Corpus-based speech synthesis principle is used in many TTS systems [7]. According to the next-generation TTS [8], systems are needed to work with speaking styles and emotions. The quality of Text-to-Speech systems is enhancing day by day with that the application field of TTS is also escalating swiftly. TTS system is becoming more suitable for everyday use to the common users, as it is also now affordable. Some uses of Text-to-Speech system are [9]:

- Talking books
- Aid to handicapped
- Education
- Games
- Telecommunication
- Man–machine communication
- Multimedia

In the digital world, information is accessible to the people who can read and understand a particular language. But people with visual impairment are also an integral part of the society. Due to their disability, they are not able to read and access the digital information. Text-to-Speech (TTS) conversion system will play very important role for physically disabled people with visual impairment. It will help visually impaired people to read the content by hearing it.

Gujarati script [10] is derived from Devanagari script. The Gujarati character set [11] contains overall 75 recognized shapes and distinct legitimate, which includes 59 characters and 16 diacritics. Fifty-nine characters are further divided into 36 consonants (2 compound and 34 singular) means ornamented sounds, 13 vowels (pure sounds), and 10 numerical digits. Sixteen diacritics are divided into 13 vowel and 3 other characters. The alphabet is ordered by logically grouping the vowels and the consonants based on their pronunciations [12].

In Western India, Gujarati is a phonetic language [13]. In Gujarati script, each character represents a syllable and is written from left to right. The consonants are called Vyanjan, and vowels are called Swar. Corresponding to each vowel, Gujarati language contains set of special modifier symbols that are attached to consonants to change their sound. These symbols are called Maatras. Modifiers occur in different shapes and are attached at the top, at the bottom right, or at the bottom part of consonant depending on the consonant. A character is conjunct if two half consonants are joined. So, characters in Gujarati can be the combination of consonant, vowels, and diacritics. (Characters in Gujarati language are shown in Fig. 1).

Aksharas are referred as the basic units of writing system. The characteristics of Aksharas are as below [14]:

1. In Indian language, Akshara is the orthographic representation of a sound in speech.
2. Aksharas are syllabic in nature.
3. The Akshara can take various forms such as V, CV, CCV, and CCCV. It can be generalized as C * V.

The overview of some of the research work related to the conversion of various types of scripts to speech is as follows:

અ આ ઇ ઈ ઉ ઊ એ ઐ ઓ ઔ અં અઃ

ા િ ી ુ ૂ ે ૈ ં ઃ

ક ખ ગ ઘ ઙ ચ છ જ ઝ ટ ઠ ડ ઢ ણ ત થ

દ ધ ન પ ફ બ ભ મ ય ર લ વ શ ષ સ હ

ળ ક્ષ જ્ઞ

૦ ૧ ૨ ૩ ૪ ૫ ૬ ૭ ૮ ૯

Fig. 1 Gujarati characters and digits

Literatures on Text-to-Speech system found in Hindi language are as follows:

Choudary [15] has worked for Hindi language. Author described a rule-based grapheme to phoneme mapping. Author has also described algorithms for the three most important subproblems in Hindi phonology, i.e., marking the syllable boundaries, schwa deletion, and pronunciation of the diacritical marks ‘ं’ (anusvara). Mishra and Shukla [16] presented methodology, application area, and some results obtained to convert text to audio in Sanskrit and Hindi. System has two main modules: Teaching and Evaluation. Their system is capable of teaching Sanskrit language with the help of Hindi language. Kabra et al. [17] provided the solution of schwa deletion while converting grapheme into phoneme for Hindi language.

Literatures found for Text-to-Speech system of English language are as follows:

Klatt [18] has developed real-time Text-to-Speech system. Ordinary English words and/or simple numerical and algebraic expressions are given as input to the system. With the help of synthesis-by-rule program and formant synthesizer, the resulting phonemic representation is converted to speech. Al-Rehili et al. [19] have discussed benefits, analysis, design, and testing of a desktop application that is able to convert English text to Arabic text. It also pronounced those texts—recognizes the English speech to convert it into a corresponding English text. It helps the user with special needs to complete their task, as the application is able to convert Text-to-Speech. They got satisfactory result.

Research works carried out in other languages are as follows:

Davaatsagaan and Paliwal [20] described TTS for Mongolian language. Authors have used general speech synthesis architecture of Festival. The system is based on diphone concatenative synthesis, applying time-domain pitch synchronous overlap add (TD-PSOLA) technique. Wolters [21] has worked for Scottish Gaelic and presented Text-to-Speech system based on a diphone. The system converts orthographic text input into speech output. The system is made up of two parts: automatic phonetic transcription module and speech synthesis module. Dika et al. [22] have worked on Albanian language. Author has given basic principles to design a system to synthesize speech from written text. They considered most repeatedly used words, two-letters, and letters for textual database. Along with the textual database, acoustics files are also included that can be used during the generation of speech. Molakatala et al. [23] have used image recognition technology with speech synthesis technology to develop a cost-effective, user-friendly image-to-speech conversion system for Telugu people. Numeric text information is converted into speech by using speech synthesis tool to speak content. With urmbookman font, they got 100% accuracy.

Some researchers have discussed various techniques and also performed comparative study. Patra et al. [24] presented a method for Text-to-Speech conversion system using MATLAB by simple matrix operations. The method uses very less amount of memory and is simple to implement. Trilla [25] has described the use of natural language processing (NLP) techniques for the generation of speech from an input text and also described the reverse process which is the generation of written text transcription from an input voice. Researcher has also described a rule-based and data-driven approach for solving speech synthesis problem. Sitaram et al. [26]

have thought about the case where there is a single speaker database but have no standardized way to write transcriptions. To address this scenario, they proposed an approach that allows them to bootstrap synthetic voices purely from speech data. Rao et al. [27] have addressed Indian languages and explained the design of a syllable-based concatenative waveform synthesizer. As Indian languages are made up of syllable, a syllable-like unit is taken into consideration. Raj et al. [14] have discussed the issues related to building Text-to-Speech systems for Indian languages. The issues addressed are pronunciation rules for Aksharas, font-to-Akshara mapping, and text normalization. Onaolapo et al. [28] have also explored Text-to-Speech system by explaining digital signal processing (DSP) module and natural language processing (NLP) module. Kishore et al. [29] have presented brief overview of unit selection in speech synthesis and issues relevant to the development of voices for Indian languages. Balajthy [30] provided an overview of the TTS technology and its application, and then provided the summary of the research on benefits of TTS for struggling readers. Sasirekha and Chandra [2] have described a tutorial on Text-to-Speech by providing summary of the published literature, and Gupta and Kumar [9] have given a comparative study of Text-to-Speech system for Indian languages.

Adequate amount of literature is found in many languages and it is able to convert Text-to-Speech, but so far, we have not come across the work in Gujarati Text-to-Speech system. So we aim to work in the conversion of digital Gujarati Text-to-Speech system that can help visually impaired people to read the content. Developing TTS system is an intricate process and it includes following challenges:

- It is difficult to identify proper syllable units that can match with the written text. Sometimes there are hidden vowels in the words that also need to be identified.
- Conversion is difficult, as we also need to consider pronunciation, punctuations, and white spaces that affect the rhythm during the speech.
- Through TTS system, sometimes it becomes difficult to produce semantic representations of input text; as a result, a variety of techniques are used to presume the proper way to disambiguate homographs, like examining neighboring words.
- It is difficult to provide naturalness and intelligibility features to speech synthesis system. The ideal Text-to-Speech system should be both natural and intelligible.

Intelligibility—The ease with which the outcome is understood
Naturalness—How much close the output sounds like human speech.

2 Conversion Tool

The implementation methodology is divided into two main parts: First, text processing—It is carried out on Gujarati language text and second, speech generation—Speech is generated according to the Gujarati text given to the system by considering the speech database.

2.1 Speech Database

Text processing and natural language processing (NLP) [25] technique are used to produce audio file from the input text. To produce audio for the text, we need a speech database. Speech database contains audio file of each Gujarati characters such as digits, vowel, and consonant. It also contains audio of Akshara, i.e., combination of consonant and vowel that contains compound characters and joint characters.

Speech database is maintained in the form of file system. For all the character and its Akshara, separate folders are maintained. So at the time of searching the audio of particular character, number of matches are reduced and it will upgrade the performance.

For example, UTF-8 code of સ is 2744. So folder is created with name 2744 and within that all the Aksharas of સ are stored as a separate file. That is, 274427652741 is સ્વ, 2744276527412750 is સ્વા, and so on.

2.2 Mapping Technique

For the conversion of Gujarati text into speech, mapping technique is used. As we want to convert digital Gujarati text, i.e., in UTF-8 format, characters can be identified with the help of UTF-8 code.

The file that contains digital Gujarati text is read character by character. According to the character encountered, it is determined that individual character is to be matched or combination of consonant and vowel is to be matched. So if combination of consonant and vowel, i.e., Akshara is to be matched, then according to the consonant code its folder name is matched and that folder is selected for further matches.

Once the folder is matched, then according to the Akshara its files are matched. And the file whose name is matched with the Akshara code is selected to play. So due to the speech database format and matching technique, only 10% database is used for matching to identify the character, which upgrades the performance of the system.

2.3 Algorithm

An algorithm is developed to convert digital Gujarati text into speech. UTF-8 is the format used worldwide for creating digital text in various languages, so we have used text file in UTF-8 format. During the conversion, system works along with the input file in UTF-8 format character by character. According to the character, mapping is done as explained in topic 2.2 with the corresponding audio file present in the speech database as explained in topic 2.1, and then the Akshara is spoken out.

The steps of the algorithm are as follows:

Step 1: Extract the characters from the file containing digital Gujarati text (UTF-8 format).

Step 2: Convert all the characters to its UTF-8 code.

Step 3: Read code one by one and check it.

Step 4: According to the combination of the vowel and consonants, find the audio file corresponding to the character and read that file (only Gujarati letters along with compound characters and numerals).
If the code is for consonant, then check the following code if it is of vowel then

Combine both the characters and according to that find the audio file.

Else if the code is for consonant and the following code is also of consonant then

Find the audio file for the previous one character code.

Else if the code is for consonant then check the following character if it is the half character identifier then

Again check the next following character code if it is consonant then

Retrieve the next character code if it is the consonant then

Combine previous three codes and find the audio file for the character.

Else if the next following character code is of vowel then

Combine all previous four codes and find the audio file for it.

Step 5: If the code is of the digits (0–9) then directly that digit code is matched with the speech database and corresponding audio file is selected.

Step 6: The audio file that is found after the mapping, all files are combined with each other to form a single audio .wav file.

Step 7: The .wav file that is generated can be played and is also saved for the future use.

This algorithm is totally based on text processing and mapping technique.

3 Testing and Result

The system is tested with Gujarati texts that are gathered from various regional newspapers Web sites such as Sandesh [31], Divya Bhaskar [32], and Gujarat Samachar [33]. It shows that algorithm works perfect with all Gujarati characters along with compound characters and digits and gives satisfactory result. But there is still scope for improvement in the execution time and increase in the performance of the system.

4 Conclusion

The foremost rationale of the work was to develop Text-to-Speech system that assists visually impaired people. The system uses mapping technique that maps digital Gujarati text containing compound words and numerals to audio file. The audio file selected according to the mapping is concatenation to form a single audio file that can be heard by the visually disabled people. The proposed work is tested on the texts that have been collected from different regional newspaper Web sites. This system will be very useful for the visually impaired people as they will be able to hear all online documents in their regional language present in digital form.

Acknowledgements We would like to thank trustee Shri. Anand Chokhavala, the Principal Mrs. Manisha Gajjar, and the recording studio In-charge Ms. Jyoti Jariwala of Ambaben Maganlal Andhjan Shala, Surat, for their cooperation by supporting us for recording audio and by providing necessary resources needed for the work. We would also like to thank Dr. Naren Burade to motivate us for the work on Text-to-Speech conversion for visually impaired people.

References

1. Speech, http://dictionary.reference.com/browse/speech
2. Sasirekha, D., Chandra, E.: Text To speech: a simple tutorial. Int. J. Soft Comput. Eng. (IJSCE) **2**(1), 275–278 (2012)
3. What is Diphone? http://www.wisegeek.com/what-is-a-diphone.htm
4. Syllable, http://www.thefreedictionary.com/syllables
5. Text, http://whatis.techtarget.com/definition/text
6. Patil, H., Patel, T., Talesara, S., Shah, N., Sailor, H., Vachhani, B., Akhani, J., Kanakiya, B., Gaur, Y., Prajapati, V.: Algorithms for speech segmentation at syllable-level for text-to-speech synthesis system in Gujarati. In. Proceeding of International Conference on Asian Spoken Language Research and Evaluation, pp. 1–7 (2013)
7. Kumar, R., Kishore, S., Gopalkrishna, A., Chitturi, R., Joshi, S., Singh, S., Sitaram, R.: Development of Indian language speech databases for large vocabulary speech recognition systems. In: Proceedings of International Conference on Speech and Computer (2005)
8. Black, A., Zen, H., Tokuda, K.: Statistical parametric speech synthesis. In: Proceeding of ICASSP, vol. 4, pp. 1229–1232 (2007)

9. Gupta, S., Kumar, P.: Comparative study of text to speech system for indian language. Int. J. Adv. Comput. Inf. Technol. 199–209 (2012)
10. Baheti, M., Kale, K., Jadav, M.: Comparison of classifiers for Gujarati numeral recognition. Int. J. Mach. Intell. **3**(3), 160–163 (2011)
11. Suthar, B.: Gujarati-English learner's dictionary, http://ccat.sas.upenn.edu/plc/gujarati/guj-engdictionary.pdf
12. Kayasth, M., Patel, B.: Offline typed Gujarati character recognition. Nat. J. Syst. Inf. Technol. **2**(1), 73–82 (2009)
13. Sojitra, B., Dhakad, V.: Neural network in character recognition of Gujarati script. J. Inf. Knowl. Res Comput. Eng. **2**(2), 269–272 (2012)
14. Raj, A., Sarkar, T., Pammi, S., Yuvraj, S., Bansal, M., Prahallad, K., Black, A.: Text Processing for text-to-speech systems in Indian languages. In: ISCA Workshop on Speech Synthesis, pp. 188–193 (2007)
15. Choudhury, M.: Rule based grapheme to phoneme mapping for hindi speech synthesis. In: 90th Indian Science Congress of the International Speech Communication Association-ISCA (2003)
16. Mishra, P., Shukla, J.: Research proposal paper on Sanskrit voice engine: convert text-to-audio in Sanskrit/Hindi. Int. J. Comput. Appl. **70**(26), 30–34 (2013)
17. Kabra, S., Agarwal, R., Yadav, N.: Rule based Schwa deletion algorithm for text to speech synthesis in Hindi. In: Advanced Computing, Networking and Informatics, Springer, vol. 1 (2014)
18. Klatt, D.: The Klattalk text-to-speech conversion system. In: Acoustics, Speech and Signal Processing IEEE International Conference on ICASSP, pp. 1589–1592 (1982)
19. Al-Rehili, A., Al-Juhani, D., Al-Maimani, M., Ahmed, M.: A Novel approach to convert speech to text and vice-versa and translate from English to Arabic language. Int. J. Sci. Appl. Inf. Technol. **1**(2), 57–64 (2012)
20. Davaatsagaan, M., Paliwal, K.: Diphone-based concatenative speech synthesis system for mongolian. In: Proceeding of International Multi Conference of Engineers and Computer Scientists, vol. 1 (2008)
21. Wolters, M.: A Diphone-based Text-to-speech system for Scottish Gaelic, Thesis (1997)
22. Dika, A., Maxhuni, A., Rexhepi, A.: The principles of designing of algorithm for speech synthesis from texts written in Albanian language. Int. J. Comput. Sci. Issues **9**(3), 175–180 (2012)
23. Molakatala, N., Kumar, M., Bhaskar, U.: Image to speech conversion system for Telugu language. Int. J. Eng. Sci. Innovative Technol. **2**(6), 161–166 (2013)
24. Patra, T., Patra, B., Mohapatra, P.: Text-to-Speech conversion with phonematic concatenation. Int. J. Electron. Commun. Comput. Technol. **2**(5), 223–226 (2012)
25. Trilla, A.: Natural language processing techniques in text-to-speech synthesis and automatic speech recognition (2009)
26. Sitaram, S., Palkar, S., Chen, Y., Parlikar A., Black, A.: Bootstrapping Text-to-Speech for speech processing in languages without an orthography. In: Proceeding of ICASSP International Conference, pp. 7992–7996 (2013)
27. Rao, M., Thomas, S., Nagarajan, T., Murthy, H.: Text-To-speech synthesis using syllable like units. In: National Conference on Communication, pp. 227–280 (2005)
28. Onaolapo, J., Idachaba, F., Badejo, J., Odu, T., Adu, O.: A simplified overview of text-to-speech synthesis. In: Proceeding of World Congress on Engineering, vol. 1 (2014)
29. Kishore, S., Black, A., Kumar, R., Sangal, R.: Experiments with unit selection speech databases for indian languages. In: Proceedings of National Seminar on Language Technology Tools: Implementations of Telugu (2003)
30. Balajthy, E.: Text-to-speech software for helping struggling readers. Int. J. Read. Assoc. **8**(4) (2005)
31. Sandesh Newspaper, http://www.sandesh.com/
32. Divya Bhaskar Newspaper, http://www.divyabhaskar.co.in/
33. Gujarat Samachar Newspaper, http://www.gujaratsamachar.com/

Hidden Markov Model for Speech Recognition System—A Pilot Study and a Naive Approach for Speech-To-Text Model

S. Rashmi, M. Hanumanthappa and Mallamma V. Reddy

Abstract Today's advancement in the research field has brought a new horizon to design the state-of-the-art systems that produce sound utterance. In order to attain a higher level of speech understanding potentiality, it is of utmost importance to achieve good efficiency. Speech-to-Text (STT) or voice recognition system is an efficacious approach that aims at recognizing speech and allows the conversion of the human voice into the text. By this, an interface between the human and the computer is created. In this direction, this paper introduces a novel approach to convert STT by using Hidden Markov Model (HMM). HMM along with other techniques such as Mel-Frequency Cepstral Coefficients (MFCCs), Decision trees, Support Vector Machine (SVM) is used to ascertain the speakers' utterances and catalyse these utterances into quantization features by evaluating the likelihood extremity of the spoken word. The accuracy of the proposed architecture is studied, which is found to be better than the existing methodologies.

Keywords Bayesian classifier · Context-dependency · Decision trees
Classification · Hidden Markov Model (HMM) · Phonetics
Support Vector Machine (SVM)

S. Rashmi (✉) · M. Hanumanthappa
Department of Computer Science and Applications, Bangalore University,
Bangalore 560056, India
e-mail: rashmi.karthik123@bub.ernet.in

M. Hanumanthappa
e-mail: hanu6572@hotmail.com

M. V. Reddy
Department of Computer Science, Rani Channamma University,
Vidyasangam, Belgaum 591156, India
e-mail: mallammantreddy@gmail.com

S. S. Agrawal et al. (eds.), *Speech and Language Processing for Human-Machine Communications*, Advances in Intelligent Systems and Computing 664,
https://doi.org/10.1007/978-981-10-6626-9_9

1 Introduction

Speech is the flow of thoughts in the form of natural language which is produced by articulating the sounds that are generated. Speech includes the formation of words and sentences. Perhaps speech is a perfect blend of rhythm and prosody, and hence, Concatenative Speech Analysis (CSA) has become extremely popular [1]. The primary target of CSA is to produce the phonetic structures and prosody models for the speech.

Speech-to-Text (STT) is a computer-based system that enables the user to enter the data in the form of speech, and then, it is converted into the textual form of data. Such a process automatically works without the human intervention. Over the past few decades, there is a tremendous amount of improvement in this arena, and it is becoming famous commercially as well. However, STT systems demand high quality, precision and accuracy. The coherence of STT mainly depends on the vocabulary size, speaker dependent versus independent, algorithms used, rate of speech and various other language constraints, and thus, its accuracy varies from system to system. This research paper focuses on studying the phonetic models and its components. We also aim to develop an accurate STT synthesizer by applying Data Mining and Natural Language Processing techniques in order to achieve improved efficiency as compared to the existing STT systems.

Speech is characterized by its temporal structure rather than spatial features; henceforth, speech always results in spectral vectors that span the audio frequency range. Furthermore, speech is characterized by the statistical models. In the persistence of the above-said fact, Hidden Markov Model (HMM) is a powerful framework that helps to construct the sound structure models more efficiently and effectively. HMM is considered as one of the substantial technique that is bound within every modern speech recognition system. It is because of this fact that it can be called as heart of the speech synthesizer systems. In the upcoming sections, evolution of STT, architecture of STT using HMM-based recognizer, implementation mechanisms and various challenges for this implementation have been discussed.

2 Literature Survey

The HMM is a famous decision-making technique that is most widely used in speech recognition systems. The available speech synthesizers using HMM are ATRECCS and TC-STAR. However, these incur a lot of time and expense. The history of speech synthesizer way dates back to 2002, and the final output was released in the year 2005 and was working for three languages: English, Spanish and Mandarin. The speech rate was 10 Hz, and the recording precision was set to 96 kHz/24 bit. In the year 2004, one more speech synthesizer was developed and was named as ‘Blizzard challenge’. This could pronounce 1200 phonetics utterances each having 1.5 Hz. Over the years, there has been a lot of improvement in this field. All such inventions are the major motivation for this work. The current work is focusing on STT by applying HMM techniques [2]. The core subject of HMM is to estimate the probability of word sequence which is achieved with the help of huge training set text. By maximizing the probability of the feature quantization vector series of the phonemes, the recognition hypothesis will be made.

The agglomerative clustering procedure for generating the text for multiple phonemes is explained here

- Initiate the HMM synthesizer for each pair of phone
- By this, a cluster of phonemes is formed
- Search for the phone-pairs which are closely related and merge together
- Look for the phonetic dictionary for the phoneme match
- Repeat the above steps for every word in the cluster

The current work is divided into three modules. First, the characteristics of the acoustic models are studied. Among them, we have chosen prosody and rhythm. On the other hand, the second module describes the construction of phonetic dictionary mitigating the issues related to this and finally, the third module discusses the feature selection approach for language identification.

3 Architecture of STT Using HMM-Based Speech Recognizer

The proposed architecture is shown in Fig. 1. It also shows the primary ingredients of a speech identification system. The principle aim of this research paper is to examine and analyse the core structure of STT and then describe the various milestones to achieve the state-of-the-art accomplishment. This is attained by using HMM. The acoustic models of the different variants of speech input are put forward.

The input for the system is the audio waveform. The input can be any recorded speech or the recorded voice using a microphone. The wave structure of the input audio is transformed into a series of fixed size vectors which are characterized by the acoustic features. This process is called as Quantization/Feature Extraction. Next is the decoder. The decoder is distinguished by three components: (1) language identification, (2) speech/acoustic models and (3) pronunciation dictionary. Furthermore, the decoder aims at identifying the words that are most likely to be indicated by the feature vectors, i.e. decoder produces the pronounced word as shown by the following equation (Eq. 1).

$$\hat{w} = [\max\{P(w/X)\}]_{[m\hat{X}n]} \tag{1}$$

where w = words $\{w_1, w_2 \ldots w_n\}$, X = Feature vector $\{X_1, X_2, X_n\}$.

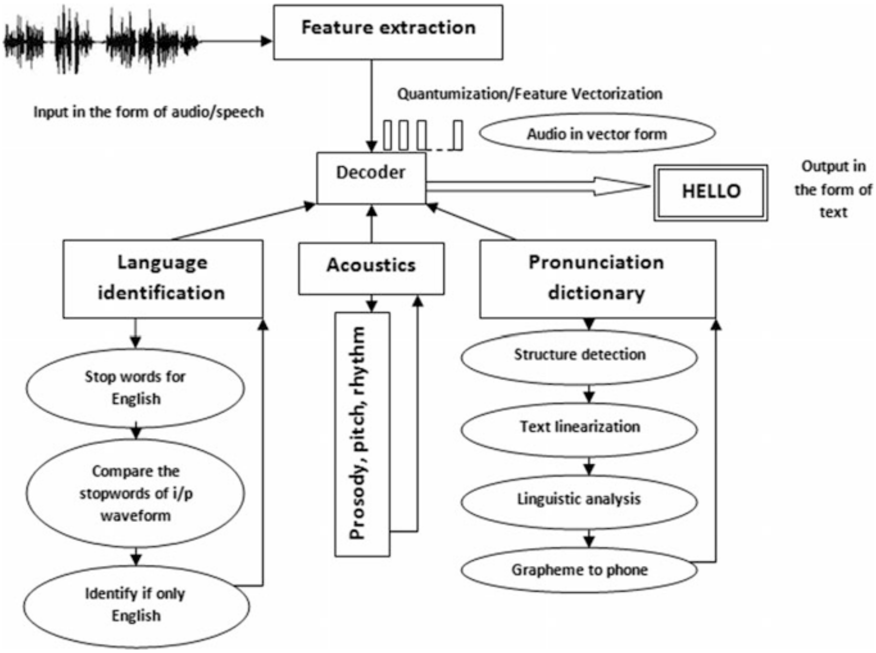


Fig. 1 Architecture of Speech-to-Text using HMM

However, the probability of $P(w/X)$ is extremely tough to predict by using a brute force strategy. Therefore, by using a Bayesian transformation rule, Eq. 1 can take the equivalent form which is much easier to find the solution.

$$\hat{w} = \sum_{x=1}^{N} \frac{\max[P(w/X)]}{p(w).p(x)}.p(w).p(x)]_{(mXn)} \tag{2}$$

The likelihood of the probabilities shown in Eq. 2 is designated by using the acoustic model in the form of phonemes. Phonemes are the basic language unit, each of which is represented in the language model. Phonemes are composed of ‘phone’, a single unit in phoneme. Such phone represents the association of a gigantic phoneme structure. For an instance, consider the word ‘beautiful’. This word is composed of four phones ˈbjuː-/tɪ-/fʊl,-/f(ə)l. There are about 45 such distinguished phonemes in English dictionary. The phonetic structure of a spoken word can be generated by concatenating all the phonemes. Since the conversion of every grapheme (written form) into the equivalent phoneme (spoken form) is based on its antecedent, the phoneme model is considered as N-gram model where the output of nth level is dependent on the $N - 1$ predecessor.

In the following section, the paradigms of the above components are explained in detail.

3.1 Feature Extraction/Quantization

A novel representation of the speech with the appropriate wave form is put forward. The major challenge in this realm is to hold the meaning of the word from getting lost during the intermediate conversion. Feature vectors are accomplished using one of the encoding schemes, Mel-Frequency Cepstral Coefficients (MFCCs). MFCCs are a famous and a standard encoding scheme typically used for large audio files with distinct parameter fluctuation in terms of bit rate and sampling rate [3]. The speech input signal is divided into window modelling where the size of the window ranges with the size of the word and falls between 10 and 25 ms. With this, the discrete Fourier transformation is computed which is given by

$$y(f) = \sum_{N=0}^{N-1} w(m)x(m)\exp(i2\theta f(m/N)) \tag{3}$$

where N is the overall length of the window and m is the length of one discrete-time signal and f represents frequency that varies between $0\ldots N$. Then, the word length $[w(x)]$ and magnitude of every word corresponding to the time signal $[x(m)]$ are calculated logarithmically by using Mel Filter, giving

$$y'(\theta) = \ln[\sum_{N=0}^{N-1} (|y(f)|).M(p, \theta)] \tag{4}$$

This equation is nonlinear for the predefined frequencies. The end result of the quantization process is a sequence of feature vectors whose dimensionality is almost decorrelated. When such feature vectors are concatenated in an orderly fashion, we arrive at delta and theta parameters [4]. These parameters make a heuristic attempt for finding regression coefficients. Therefore, Δx_t^v, the delta parameter is evaluated by the following equation,

$$\Delta x_t^v = \frac{\sum_{i=1}^{N} w_i(\mathrm{x}_{t+i}^v - y_{t-i}^v)}{\pi \sum_{i=1}^{N} \theta(w_i)} \tag{5}$$

θ in Eqs. 3, 4 and 5 represents the angular velocity of the movements of feature vectors.

3.2 Decoder

Once the feature vectors arrive at the decoding segment, these vectors are distributed in a nonlinear manner across the speech spectrum. As noted earlier, for any

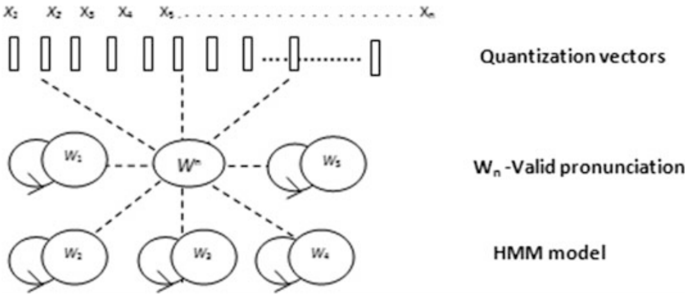


Fig. 2 Transition from feature vectors to word mapping using HMM-based phoneme model

word '*w*', there are a series of sound model produced called phonemes. Let these series be named as R_w. The probability of such likelihood can be expressed as Eq. 1. By using Eq. 1, the overall genuine and correct pronunciation can be tractable using the following equation

$$P(R/w) = \sum_{i=1}^{L} p(w/R_w), w^n \tag{6}$$

w^n indicates a valid pronunciation. The transition of such probabilistic measure is described by HMM using a transition diagram. This resolves the round boundary set R_w making the transition from its present state to all the values of w^n, for each value of n.

Furthermore, from Fig. 2, it is evident to make the following discussions.

- w^n can be generated with the help of all the independent values of w
- w^n is independent of *X;* however, the values of w^n are correlated with the value of X
- Many feature vectors can be discarded considering as noise, which also includes the millisecond gap during the intermediate phones generated by the audio input
- The order of sound utterance must be preserved

The partitioning of feature vectors into the phonemes is a major concern as its distributions are dependent on the likelihood of w's and in turn X's. Such an approach demands a high-level context-dependent covariance which is commonly referred as Beads-On-A-String (BDAS) [5]. This is because all the combinations of a valid pronunciation arrive at the interval of w^n by concatenating the sequence of w together. This imposes a large degree of context-dependency. For example, observe the value for w^n, Loot, School, Wool and Reel. The repetitive letters 'oo' or 'ee' have to be pronounced though it is same yet differs when one of them is omitted. The mapping of context-dependency is demonstrated in Fig. 3. The figure uses the conventions where W is the word spoken/input, Q represents the quantization vectors, P denotes the phonemes, L is the language representation and R is

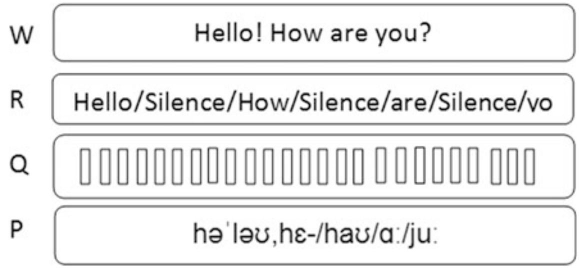


Fig. 3 Formation of phoneme modelling using content dependency structure

the logical modelling. The output dissemination can be constructed by using Gaussians distribution. According to Gaussians [6],

$$f(x/\mu_X.\sigma_X) = \prod_{x=1}^{n} \sigma_x\sigma_y \frac{1}{\sigma_X\sqrt{2\pi}} \int_{x=1}^{n-1} \mathrm{e}^{\frac{(x-\mu_X)^2}{2\sigma_X^2}} \tag{7}$$

In Eq. 7, μ_X is the mean and σ_X represents variance. This equation gives the distribution of feature vectors for the normal deviation. This is diagonal in nature.

The feature vectors result in a series of phonetic transcription by word-to-word mapping, and these are then plotted in a look-up table. This table contains the phonetic word dictionary [Logical model]; finally, the phonemes are translated to English words [7]. The association between the logical and physical model is bound together through the states of the transition. Such transitions require the usage of decision trees for every phone that has been formulated using the above-said mechanism. All the phonemes are tied at the root nodes combining the value of each phone for the state 'i' of which the nodes are later chopped into various levels until the leaf nodes. This is a greedy approach, and it is iterative in nature. Figure 4 illustrates the decision tree for this greedy approach.

In this figure, an example of nasal sounds is shown. In English, the sounds of /m/, /n/, /ng/ are nasal, produced by generating the airstream through node; the example words 'bringing' and 'hanging' as 'bri-ng + ing' and 'ha-ng + ing' are shown in the form of decision tree in Fig. 4.

From the observation made using HMM recognizer, the Bayesian Classifier describes the topology and its construction. This is depicted in Fig. 5. The following HMM topology notations are assumed. The circle represents the discrete variable and empty circle stands for loss of phonemes, whereas filled circle shows the transition of phonemes that are to be considered. Square gives the continuous values for the phone structure, empty is for 'no' in the decision tree and 'yes' is given for a filled square. The triangle shows the constraint satisfaction and conditional transition. HMM contains many hidden states. A state is said to be hidden because when we traverse through the HMM synthesizer, these states will make the transition from hidden to visible. The number of states in a HMM model depends on the sequence of tokens in the input string. This grows recursively when the

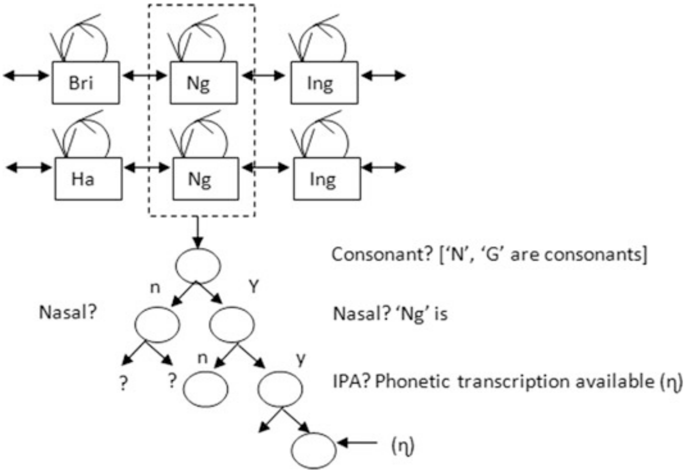

Bri
Ng
Ing
Ha
Ng
Ing
Consonant? ['N', 'G' are consonants]
n
Y
Nasal?
Nasal? 'Ng' is
n
y
? ?
IPA? Phonetic transcription available (ŋ)
(ŋ)

Fig. 4 Transition from feature vector to word mapping using HMM-based phoneme model

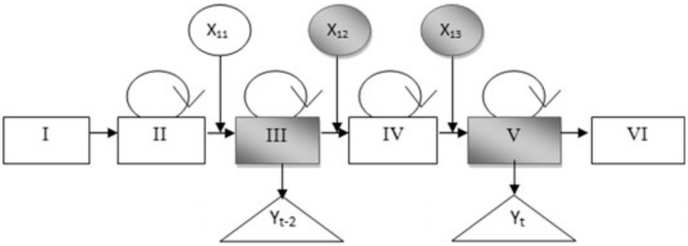


Fig. 5 Bayesian network for HMM topology

speech data increases. The lower bound on the number of states should be at least one (minimum requirement for the speech to be converted into text); however, there is no upper bound as this significantly grows with the input.

3.3 Pronunciation Dictionary

Predominantly, all the speech recognition system uses corpora which contain the phonetic transcriptions for the words of the native language. Such corpora are called the phonetic dictionary. This forms the training data. Nevertheless, even a clearly defined lexicon fails to provide the phonemes for all the pronunciations made by human. Besides, if an attempt is made to provide a dictionary which contains all such phonemes, then the size of the dictionary will be excessively large. Support Vector Machine (SVM) along with the rule-based classifier across the phoneme model is proven to be much coherent for devising the phonetic corpora with not much modification in the dictionary. Using SVM, the phonemes model will generate new paths, and this is to automate the creation of new phonemes. The phonetic dictionary can be implemented in two ways: (1) dictionary-based and (2) rule-based classifier based on the SVM.

3.3.1 Dictionary Based

In this method, all the words and the corresponding phonemes are gathered. However, this could lead to a vast dictionary. This method has one more drawback, when a new word is encountered which is not found in the dictionary the output is not rendered.

3.3.2 Rule-Based Classifier

It was shown by Swamy [8] that 70% of English words can be deduced by using a subset of 2000 words only. This forms the base for our hypothesis, and henceforth, a phonetic dictionary was constructed consisting of 2500 randomly chosen, basic yet key words in English language. However, for a new word, the dictionary is trained using one of the supervised learning methods called as Support Vector Machine (SVM). With thorough literature survey, it was discovered that SVM is an optimal approach to implement phonetics. SVM, a classification technique, requires a trained/labelled data using which it categorizes a hyper plane that is favourable and optimum. The first step in SVM is to draw a line that linearly separates the points on the plane. In the next step, draw a line of equal distance between two boundaries where the line was linearly separated. This line should not be too close to the samples. If so, the points on this line will be eliminated as noise, and the related phones are abandoned. All the labelled samples that fall on an optimal linear bar form the support vectors. If the line is not 'linearly separable', then it is called as 'perceptron'. Assume a labelled data across M and N coordinates such that M_i and N_i are given by 1, 2,… Z. $M \in E$ where E is the edit distance that calculates the level of similarity by calculating the number of edits needed to transform one text into another. $N \in -1$ to $+1$, this provides the scope of the feasible efforts to bring the required phonemes for a given input. Hence, the function $f(M, N)$ is given as $f(M, N)$ ➔ Case i: ≥ 0 (N as positive coordinates). Case ii: <0 (N as negative coordinates). Accordingly, for a precise classification $f(M, N) \geq 0$ should hold true. If this classification exists, then it is named as 'Linear Separable'. This is shown in Eq. 8.

$$f(M, N) = E[(\text{Weight}_{\text{Factor}})]_M^N + e. \tag{8}$$

Here, E represents the edit distance for all the $N - 1$ points on the hyper plane; e is the noise error that is almost negligible. For example, the word ‘HALF’ contains one character ‘L’ as noise error since it is silent.

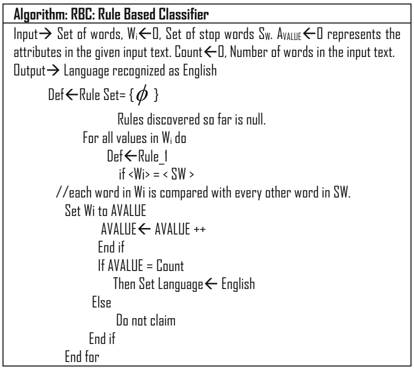
3.4 Language Recognition

The font type and the coded language have to be identified when a speech is given as an input. Therefore, the fundamental step is to identify the language spoken in the input speech.

3.4.1 Classification

A technique used to forecast the correct label for an input data is called as classification [9]. To issue loan, the bank manager must inspect the available data (training set) of a customer in order to know whether granting loan to the applicant is safe or not. Thus, a proper supervision is required to manifest a clear boundary as this technique always incurs a question of uncertainty. The likelihood of data is either they belong to a trained class or it might be rejected [10]. Classification process contains building a classifier/training data model. A list of stop words is constructed. Stop words form the basic fundamental unit which is distinctive for a language dialect (the, to, is, I, am and so on). This acts as data for the supervised learning method. The characters are compared against the training data by using IF-THEN association rules. Table 1 shows the working of proposed Rule-Based-Classifier (RBC) algorithm.

Table 1 RBC algorithm



```
Algorithm: RBC: Rule Based Classifier
Input→ Set of words, $W_i$←0, Set of stop words $S_W$. $A_{VALUE}$←0 represents the
attributes in the given input text. Count←0, Number of words in the input text.
Output→ Language recognized as English
        Def←Rule Set= {$\phi$ }
                        Rules discovered so far is null.
                For all values in $W_i$ do
                      Def←Rule_1
                        if <Wi> = < SW >
          //each word in Wi is compared with every other word in SW.
            Set Wi to AVALUE
                    AVALUE← AVALUE ++
                    End if
                    If AVALUE = Count
                        Then Set Language← English
                   Else
                        Do not claim
                  End if
            End for
```

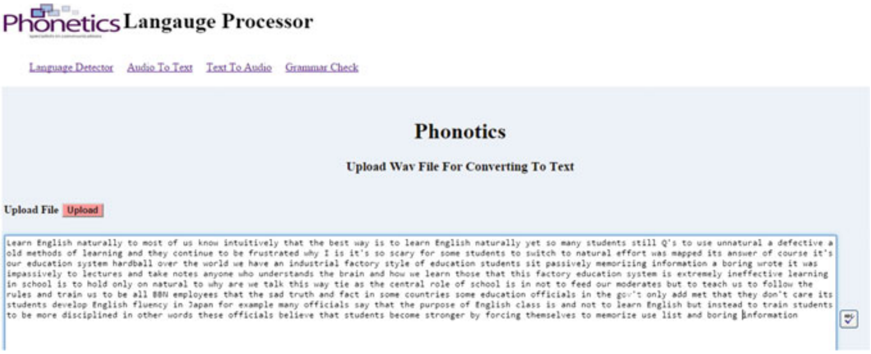
4 Implementation

The Phonetics Language Processor is an interface that the entire research project will be interpreted on. The processor includes four tabs, namely Language detector, Audio/Speech-to-Text, Text-to Audio/Speech and Grammar check. The tab, Language detector shows the identification of language as explained in the Sect. 3.4. In the second segment, Audio/Speech-to-Text is taken care off. The techniques explained in the present work are administered and executed in the fragment Audio/speech-to-Text. The third and the fourth component have been defined to address Text-to-Audio and Grammar check which is beyond the scope of this research paper.

The statistical approach for STT using HMM was showed in the earlier section. In this section, the techniques explained so far have been implemented by designing an interface using .Net platform. Figure [6] is the final output which showcases all the features explained so far. The audio/speech file is provided as input for this interface. The file has to be in .wav form only. On successful loading of file, preview button is pressed. The output in the textual form will be seen in the layout.

5 Results and Discussion

A basic strategy to obtain the phonetic transcription is by using the available morphological analyser; however, the efficiency which the analyser provides does not cross above 75% in a huge volume of dictionary which contains 300 k words. By these, we can conclude that the pronunciation must be hand built by generating rules. In order to deduce the system with such rule is a major challenge. In the present work, a small number of hand cribbed words were added to the phonetic dictionary that includes 2500 words and 1500 sentences. In order to reduce the complexity of dictionary, many variations such as 'ya, yep, gonna, wanna' were



```
Learn English naturally to most of us know intuitively that the best way is to learn English naturally yet so many students still Q's to use unnatural a defective a
old methods of learning and they continue to be frustrated why I is it's so scary for some students to switch to natural effort was mapped its answer of course it's
our education system hardball over the world we have an industrial factory style of education students sit passively memorizing information a boring wrote it was
impassively to lectures and take notes anyone who understands the brain and how we learn those that this factory education system is extremely ineffective learning
in school is to hold only on natural to why are we talk this way tie as the central role of school is in not to feed our moderates but to teach us to follow the
rules and train us to be all BBN employees that the sad truth and fact in some countries some education officials in the gov't only add met that they don't care its
students develop English fluency in Japan for example many officials say that the purpose of English class is and not to learn English but instead to train students
to be more disciplined in other words these officials believe that students become stronger by forcing themselves to memorize use list and boring information
```

Fig. 6 Interface showing the conversion of Speech-to-Text

mapped to their base forms. The phonetics has an average of 4.3 variants per word in English. This shows the importance of pronunciation dictionary which maps one-to-one modelling. In reality, this results in a huge vocabulary; hence, by adopting the HMM-based speech synthesizer, this number can be reduced.

To summarize, the acoustic speech model was studied with the help of HMM and Gaussian distributions whilst decision tree supported the assumptions drawn on these. The overall study showcases the following key features.

- Monophonic mapping was deduced by HMM–Gaussian model that calculates mean and variance of the training data. Later, these monophonic transcriptions are mapped onto the phonetic dictionary that was built beforehand as explained in Sect. 3.3.
- With each monophonic word, biphonemes, triphonemes and multiphonemes are transformed into phonetic transcription and once again re-estimated using context-dependent model structure.
- The language of all these phonemes must be in English. This language identification is done to ensure that the input audio/speech was in English. If otherwise, the interface does not provide the output. The reason for this is that the phonetic dictionary was built only for the English language.
- The output in rendered in the form of text.

The performance of the STT synthesizer was evaluated for a different range of speech input. In practice, it was found that the efficiency of the speech recognition system varies with the size of the vocabulary. When a set of audio files are interpolated as the input, the equivalent text was received as the output. Table 2 evaluates the performance of the proposed architecture. The input speech was named as S1, S2, S3, S4 and S5. Each of this input consisted of all the different type of speech variants. S4 was the size of 20 min, and it was expected to be much harder when compared to other speech files. It carried disruption such as background music and embodied other kinds of interference which included multilingual context. However, the system gave the correct results by identifying only the English language. The results achieved by our approach are fascinating and were found to be

Table 2 Result evaluation on the interface

Input	Total words	Words correctly identified	Words that are identified wrong	Words that are not identified	% correctness	Of recall (%)	Precision (%)
S1	200	195	3	2	97.5	98.1	98.3
S2	18	14	2	2	77.7	87	87
S3	570	497	38	35	87.19	93	92
S4 (English only)	8478	7023	97	358	94.63	95	98
S5	4396	3762	363	271	85.57	93	91

Table 3 Comparison of the results of existing systems

Sl. No	Features	Techniques/ Methodologies	Accuracy (%)	Drawbacks
1	MFCCs	Rule based, PRLM	82	Concentrates only on the feature extraction
2	Acoustic models	HMM	79	Generation of the acoustic features results in chopping of essential elements
3	Phonotactics, acoustic, prosodic information	Trigram model	NA	Does not show the conversion of Speech-to-Text
4	Articulation of sounds	SVM	83	Time taken for the sound apprehensions is high
5	Phonetic features, nazal, articulation features	HMM & N-gram model	78	Concentrates on the phonemes generation; however, the system does not show the performance for the huge data

88.51% accurate and efficient. Table 3 provides a comparative study of the proposed architecture with existing methodologies and algorithms.

6 Conclusion

In this research paper, STT paradigm using HMM is put forward. HMM is an excellent technique for resolving many computational language challenges in the field of speech recognition. The intention of this work was to develop an interface using the acoustic models. It was found that the output text was being trained automatically on the input speech. The various feature distribution and their effect on the output were studied at the same time. In terms of chief investigation in the STT, it was discovered that the model can be extended for multiple languages by building the phonetic dictionary of the same along with some modifications in the phonemes. HMM-based model adopts several assumptions on the feature quantization, training data and context-dependency. Conventionally, a few of those presumptions can be compromised to some extent. Finally, it should be noted that despite the advantages of HMM and its superiority, many expostulate that HMM is blemished. This is of course true under many circumstances as the system gets vulnerable with the speaking styles, frequency, dialects and accents. Perhaps there has been no good alternative for HMM and it is because of this that HMM is still undeniably the best approach for implementing STT.

References

1. Alias, F., et al.: Towards high-quality next generation text-to-speech synthesis: a multi domain approach by automatic domain classification. IEEE Trans. Audio Speech Lang. Process. **16**(7) (Sept 2008)
2. Abushariah, A.A.M., et al.: English digits speech recognition system based on Hidden Markov Models. In: IEEE Conference 2010, ICCCE. doi:10.1109/ICCCE.2010.5556819
3. Hossan, M.A., et al.: A novel approach for MFCC feature extraction. In: IEEE Conference 2010, ICSPCS. doi:10.1109/ICSPCS.2010.5709752
4. Bsyrne, W.: Minimum Bayes risk estimation and decoding in large vocabulary continuous speech recognition. IEEE **E89-D**(3), 900–907 (2006)
5. Patel, I., et al.: Speech recognition using hidden Markov model with MFCC-subband technique. In: IEEE Conference (2010). doi:10.1109/ITC.2010.45
6. Duan, W., et al.: Weighted naive Bayesian classifier model based on information gain. In: IEEE Conference, (ISDEA). doi:10.1109/ISDEA.2010.226
7. Gales, M., et al.: The application of hidden Markov models in speech recognition. Found. Trends Signal Process. **1**(3), 195–304 (2008)
8. Swamy, S., et al.: An efficient speech recognition system. Comput. Sci. Eng. Int. J. (CSEIJ) **3** (4) (Aug 2013)
9. Kholghi, M., et al.: Classification and evaluation of data mining techniques for data stream requirements. In: IEEE Conference on Computer Communication Control and Automation (3CA). doi:10.1109/3CA.2010.5533759
10. Shahrokhi, N., et al.: Targeting customers with data mining techniques: classification. In: 2011 International Conference on User Science and Engineering (i-USEr), IEEE, New York. doi:10.1109/iUSEr.2011.6150567

Speaker-Independent Recognition System for Continuous Hindi Speech Using Probabilistic Model

Shambhu Sharan, Shweta Bansal and S. S. Agrawal

Abstract In this generation of IT, communicating with machines in an expedient manner using human speech that too in their own language is highly desirable. This is achieved using speech recognition systems which allow the general public to speak to the machine by recognizing their voice. Hindi being the most widely spoken language with approx. 260 million first-language speakers [1] should have a real-time recognition system. The main objective of this paper is to develop a speaker-independent system which can recognize continuous Hindi speech in real-time scenario. This paper presents the feasibility of MFCC for feature extraction and dynamic time warping to compare the test sequence. The system has been trained on 8 h of audio data and a trigram language model trained with 30K words. With a dictionary of 6K words, the system gives a word accuracy of 80–85%.

Keywords Hindi speech recognition · Hindi ASR · Mel-frequency cepstral coefficients · MFCC · Hidden Markov Model · HMM · Pronunciation dictionary · Language model · Acoustic model · Dynamic time warping

1 Introduction

Hindi is a phonetically rich language written in Devanagari derived from the Brahmi script. Devanagari was initially used to write Sanskrit but later adapted for Hindi. Hindi consists of 11 vowels and 33 consonants as shown in Table 1. Nearly 260 million people speak Hindi natively even then Hindi barely has sufficient

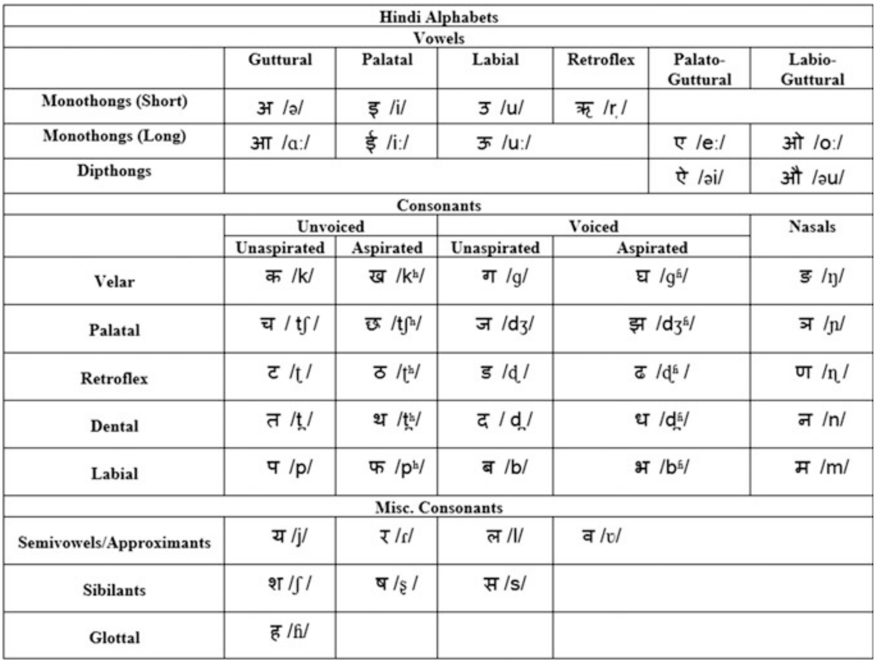
S. Sharan (✉) · S. Bansal · S. S. Agrawal
KIIT College of Engineering, Gurgaon, India
e-mail: reachshambhu@gmail.com

S. Bansal
e-mail: bansalshwe@gmail.com

S. S. Agrawal
e-mail: ss_agrawal@hotmail.com

S. S. Agrawal et al. (eds.), *Speech and Language Processing for Human-Machine Communications*, Advances in Intelligent Systems and Computing 664,
https://doi.org/10.1007/978-981-10-6626-9_10

Table 1 Hindi alphabets



Hindi Alphabets						
Vowels						
	Guttural	Palatal	Labial	Retroflex	Palato-Guttural	Labio-Guttural
Monothongs (Short)	अ /ə/	इ /i/	उ /u/	ऋ /r̩/		
Monothongs (Long)	आ /ɑː/	ई /iː/	ऊ /uː/		ए /eː/	ओ /oː/
Dipthongs					ऐ /əi/	औ /əu/
Consonants						
	Unvoiced		Voiced			Nasals
	Unaspirated	Aspirated	Unaspirated	Aspirated		
Velar	क /k/	ख /kʰ/	ग /g/	घ /gʱ/		ङ /ŋ/
Palatal	च / tʃ /	छ /tʃʰ/	ज /dʒ/	झ /dʒʱ/		ञ /ɲ/
Retroflex	ट /ʈ /	ठ /ʈʰ/	ड /ɖ /	ढ /ɖʱ /		ण /ɳ /
Dental	त /t̪/	थ /t̪ʰ/	द / d̪/	ध /d̪ʱ/		न /n/
Labial	प /p/	फ /pʰ/	ब /b/	भ /bʱ/		म /m/
Misc. Consonants						
Semivowels/Approximants	य /j/	र /ɾ/	ल /l/	व /ʋ/		
Sibilants	श /ʃ /	ष /ʂ /	स /s/			
Glottal	ह /ɦ/					

resources in speech area. Since speaking in one's own language is the easiest form of communication one can do, it is obviously desirable by the general public to communicate with machines in their own language orally. However, one of the challenges in designing a recognition system is variability in the speech, thus designing a system involves careful selection of feature extraction technique and modeling methods.

The highly accurate and large vocabulary continuous speech recognition systems in the real-world scenario are still too far and expensive. Researchers are trying hard to improve the accuracy of the speech processing techniques. In recent pasts, few researchers have come up with their developments and techniques in recognition systems for Indian languages: Agrawal et al. [2], Samudravijaya [3], Pruthi et al. [4], Neti et al. [5], Mathur et al. [6], and Kumar and Aggarwal [7], etc.

2 Corpus Creation

2.1 Data Collection

To begin with, initially the raw data were collected by crawling various Hindi resources available online [8, 9]. The raw data were then cleaned by removing the punctuation marks (|,,? etc.), special symbols (unwanted characters), converting numeric character to alphabets (e.g., 1 to एक). Furthermore, the unique words were extracted from the cleaned data, and the corpus of 100 phonetically rich sentences was created. This corpus was then used for recording purpose.

2.2 Data Recording

The recording equipment includes a desktop computer Lenovo Think Centre 1607G6Q and a head-held microphone (Shure Beta 54) with a preamplifier and M-Audio amplifier (digital interface). The recording was done using the software "Goldwave". A soundproof room was selected as the recording place where the noise frequency is very low to ensure the high-quality recording. We have invited 50 speakers—25 males and 25 females—in the age-group of 18–50. Each sentence is uttered by every speaker, and the corresponding sound file is saved as a separate wave file (.wav extension) with the sampling frequency of 16 kHz. Throughout the recording process, the speakers need to wear the microphone and utter each sentence. The recording is operated by the audio engineer. If the pronunciation went wrong or sounds odd, it is recorded again to ensure the accuracy.

3 Methodology

3.1 Feature Extraction

The first step in any speech recognition system is to extract features from which linguistic contents can easily be identified. There exists numerous feature extraction techniques such as LPCC, PLP, MFCC [10–12]. We have used mel-frequency cepstral coefficients (MFCCs) for our work as it is most often used technique to create fingerprint of sound files. Also, mel scale relates perceived frequency to the actual measured frequency (Fig. 1).

The formula to convert f Hz into mel is:

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

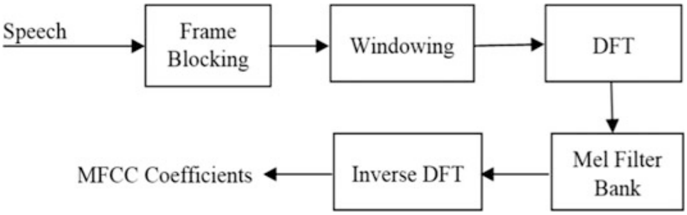


Fig. 1 MFCC process

And, from m mel to Hz is:

$$f = 700\left(10^{m/2595} - 1\right) \tag{2}$$

Initially, the continuous speech signal was framed into 25 ms frames, with frame step of 10 ms. If the frame size is smaller than the size taken, then the number of samples in the frames will not be enough to get the consistent information, and with large frame size, it can cause frequent variation in the information inside the frames. In the next step, windowing is done to decrease the interference at the beginning/end of the frame. There exist many window functions such as rectangular window, flattop window, hamming window. Here, hamming window had been implemented. The DFT basically converts each frame from time domain to frequency domain. To apply DFT on each frame of signal $S(n)$ having N samples, fast Fourier transform (FFT) is used:

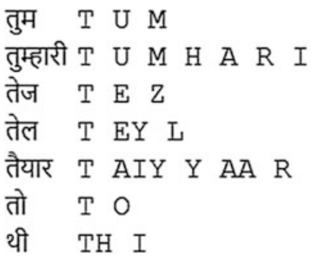
$$S_i(k) = \sum_{n=1}^{N} s_i(n)\, h(n)\, e^{-j2\pi kn/N}, \quad 1 < k < K \tag{3}$$

where $h(n)$ is an N sample-long analysis window, and K is the length of the DFT.

Then, the previously calculated spectrums are converted to mel scale, and the signal is filtered using band-pass filter. With this filter bank, it is easy to estimate the energy at any point, and the log of these energies is known as mel spectrum. Finally, inverse DFT is carried out to convert the log mel spectrum back to time domain, and the output is known as mel-frequency cepstral coefficient.

3.2 Pronunciation Dictionary

Phoneme is the basic unit of sound in any language. The pronunciation dictionary contains words and mapping to their phonetic contents as shown in Fig. 2. The words covered in the dictionary can only be recognized by the recognizer.



तुम	T U M
तुम्हारी	T U M H A R I
तेज	T E Z
तेल	T EY L
तैयार	T AIY Y AA R
तो	T O
थी	TH I

Fig. 2 Pronunciation dictionary

3.3 Acoustic Modeling

Acoustic modeling uses a statistical machine learning technique known as Hidden Markov Model (HMM) to capture the variations in speech. Here, each word is modeled as a whole, separately and is stored in a file which is then used by decoder for matching with real-time speech input (Fig. 3).

The acoustic model is created using SphinxTrain. For acoustic modeling, we first created a transcript file containing the word in the same sequence as occurred in the speech data. SphinxTrain then searches the dictionary which maps the words to phonemes.

3.4 Language Modeling

Language model captures the fundamental grammatical structure of the language using *N*-Gram model. *N*-Gram basically captures the probability of a word occurring after ($N - 1$)-Gram. We have used trigram model in our language model. At the start, word list was generated with word frequency, i.e., the number of occurrences of each word in the cleaned, collected text corpus. This is known as the unigram file. The unigram file is then sorted alphabetically, and a vocabulary file is generated out of it. Next, we create a trigram file containing the list of all possible

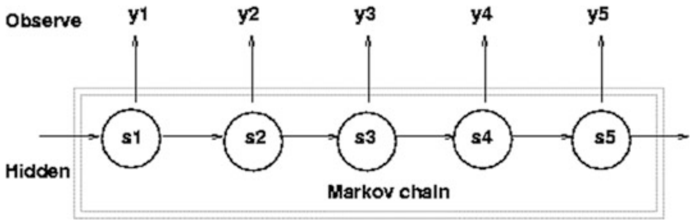


Fig. 3 Hidden Markov Model

trigram with their frequency. Further, a binary id 3-g is created based on the vocabulary, and at last, a binary format language model is generated which can also be converted to ARPA format.

3.5 Feature Matching

Dynamic time warping (DTW) was used for feature matching, i.e., alignment and comparison of input (real-time test sequence) with the stored models (reference sequence).

4 Recognition Results

The system is trained for 30K words. Then, recognition accuracy is calculated for connected words in noise-free environment using the formula:

$$\text{Recognition Accuracy} = \frac{\text{No. of Correct Words Recognized}}{\text{Total No. of Words}} \times 100$$

The overall accuracy and word correction rate for connected words is 80–85%.

Acknowledgements The assistance of Mr. A.K. Dhamija in this work is acknowledged. We also thank the management of KIIT College of Engineering, above all Dr. Harsh Vardhan Kamrah and Dr. Neelima Kamrah for providing required facilities, financial help, and support.

References

1. Information please. http://www.infoplease.com/ipa/A0775272.html
2. Sinha, S., Agrawal, S.S., Jain, A.: Continuous density hidden markov model for context dependent Hindi speech recognition. In Advances in Computing, Communications and Informatics (ICACCI), International Conference on 2013, pp. 1953–1958. doi:10.1109/ICACCI.2013.6637481 (2013)
3. Samudravijaya, K.: Computer recognition of spoken hindi. In Proceeding of International Conference of Speech, pp. 8–13. Music and Allied Signal Processing. URL http://speech.tifr.res.in/chief/publ/00hindiReco.doc (2000)
4. Pruthi, T., Saksena, S., and Das, P K.: Swaranjali: isolated word recognition for hindi language using vq and hmm. In International Conference on Multimedia Processing and Systems (ICMPS), IIT Madras. URL http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.144.4538&rep=rep1&type=pdf (2000)
5. Neti, C., Rajput, N., Verma, A.: A large-vocabulary continuous speech recognition system for hindi. IBM J. Res. Dev. URL http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.17.5089&rep=rep1&type=pdf (2004)

6. RajatMathur, Babita, and Abhishek Kansal. Domain specific speaker independent continuous speech recognizer using julius. In Proceedings of ASCNT, pp. 55–60. Noida, CDAC (2010)
7. Kumar, K., Aggarwal, R.K.: Hindi speech recognition system using htk. Int. J. Comput. Bus. Res. **2**, URL http://www.researchmanuscripts.com/PapersVol2N2/IJCBRVOL2N2P3.pdf (2011). ISSN 2229-6166
8. Gupta, R., Sivakumar, C.: Speech recognition for hindi language. Master's Dissertation. IIT BOMBAY (2006)
9. Venkataramani, B.: Sopc-based speech-to-text conversion. In Nios II Embedded Processor Design Contest Outstanding Designs, pp. 83–108. URL https://www.altera.com/en_US/pdfs/literature/dc/2006/i2.pdf (2006)
10. Aggarwal, R.K., Dave, M.: Using gaussian mixtures for hindi speech recognition system. Int. J. Signal Process. Image Process. Pattern Recogn. **2**(4). URL http://www.sersc.org/journals/IJSIP/vol4_no4/13.pdf (2011)
11. Mishra, A.N., Chandra, M., Biswas, Astik., Sharan, S.N.: Robust Hindi connected digits recognition. Int. J. Signal Process. Image Process Pattern Recog. **4**(2):79–90. URL http://www.sersc.org/journals/IJSIP/vol4_no2/8.pdf (2011)
12. Sivaraman, G., Samudravijaya, K.: Hindi speech recognition and online speaker adaptation. In International Conference on Technology Systems and Management (ICTSM), Int. J. Comput. Appl.® (IJCA). pp. 27–30 (2011)

A Robust Technique for Handwritten Words Segmentation into Individual Characters

Amit Choudhary and Vinod Kumar

Abstract Segmentation of individual characters from a scanned word image is the most critical step of a typical optical character recognition (OCR) system. A robust segmentation algorithm is proposed in this paper. The word images are segmented into individual characters after skew angle correction and the thinning process, to get the single pixel stroke width. Ligatures of the touching characters are detected by keeping in view the geometrical shape of the English alphabets. The proposed vertical segmentation technique is used to cut individual characters from the handwritten cursive words. The proposed algorithm delivers excellent segmentation accuracy when tested on a local database.

Keywords Segmentation · OCR · Word recognition · Preprocessing

1 Introduction and Historical Background

Nowadays researchers are trying to introduce human brain's intelligence and capability into a computer system to recognize the information written on paper. In an OCR system, good character recognition accuracy can be achieved if the characters in the handwritten script are well segmented. Many researchers had already achieved very good segmentation results [1], but the scope of improvement is always there and superior segmentation results are always awaited. Technological advancements during the last 40 years in the area of document and character recognition are presented [2]. A new technique to recognize handwritten as well as typewritten English text is presented [3]. It does not require the thinning process,

A. Choudhary (✉)
Maharaja Surajmal Institute (GGSIP University), New Delhi, India
e-mail: amit.choudhary69@gmail.com

V. Kumar
Delhi Technological University, New Delhi, India
e-mail: vinodkumar@dce.edu

S. S. Agrawal et al. (eds.), *Speech and Language Processing for Human-Machine Communications*, Advances in Intelligent Systems and Computing 664,
https://doi.org/10.1007/978-981-10-6626-9_11

and it delivered 80% accuracy. The slant and skew correction was not performed during preprocessing the word images by the authors [4].

2 Preprocessing Techniques and Database Preparation

To demonstrate the proposed segmentation algorithm, handwritten word samples written on colored or noisy background have been collected from 10 different persons aged between 15 and 40 years. From the collection of handwriting samples, we have selected 400 handwritten words randomly to perform the proposed experiment. Figure 1 displays few samples from the local handwritten word images database.

2.1 Image Acquisition

In image acquisition, a digital photocamera or a scanner is generally used to capture the handwritten word images, and these images are saved in .bmp or .jpg file format for preprocessing. Figure 2 shows two such image samples from the database.

2.2 Preprocessing

The main objective behind preprocessing is to remove the invariabilities existing in word images. While scanning the handwritten word images, the quality may be ruined as the noise is introduced due to dust or due to colored background.

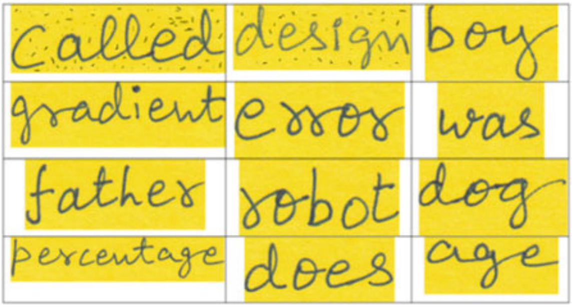


Fig. 1 Handwritten word image samples

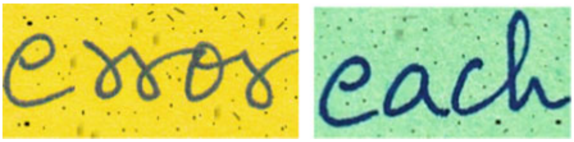


Fig. 2 Input scanned handwritten word images

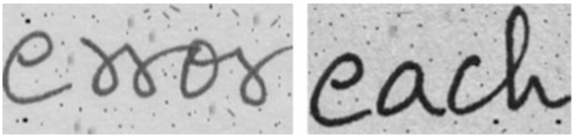


Fig. 3 Handwritten word images in grayscale format

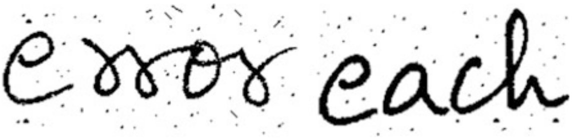


Fig. 4 Images in binary form

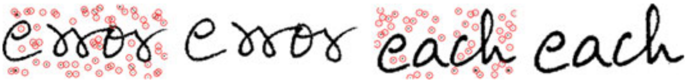


Fig. 5 Noise detection and word images after noise removal

Thresholding and Binarization.

Thresholding is necessary so that the problems can be avoided due to usage of pen of different colored ink on colored and noisy surfaces. Figure 3 shows two such grayscale images obtained after thresholding.

The grayscale images are then transformed to the binary matrix form in which a 0 represents a black pixel in the foreground and a 1 represents a white pixel in the background. Figure 4 displays such binary images.

Image De-noising and Skeletonization.

In this preprocessing stage, the noise (small foreground components and dots) induced in the image scanning process is optimally eliminated. Only the noise dots and other foreground components have been removed in this step while retaining the character components. Noise-free images are shown in Fig. 5.

As the pens of different stroke width can be used by the different writers, a lot of unevenness may exist. After the thinning process, all the handwritten word images

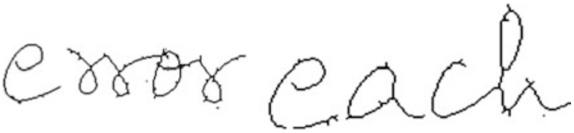


Fig. 6 Word image after thinning

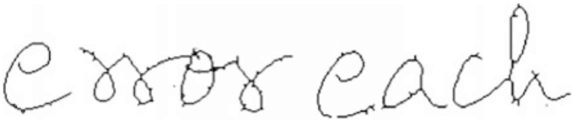


Fig. 7 Cropped image

were made to have stroke width of 1 pixel each. Such image samples following the skeletonization process are displayed in Fig. 6.

Cropping and De-skewing.
The images after de-noising are cropped to remove the extra space available around the rectangular region enclosing the handwritten noise-free word image. The skew correction is also performed on the handwritten word images. Figure 7 shows such cropped sample images after skew correction which was not performed earlier [5].

3 Proposed Segmentation Technique

The projected segmentation algorithm is designed for segmenting touching character present in the handwritten words of English language and may not work well if applied to some other languages such as Arabic or Chinese.

3.1 Overview

English language has closed as well as open characters. Closed characters have a semi-loop or a loop such as ‘g’, ‘o’, ‘p’, ‘s’, ‘a’, ‘b’, ‘c’, ‘d’, ‘e’. Open characters do not have any semi-loop or loop, e.g., ‘u’, ‘v’, ‘w’, ‘m’, ‘n’. Discriminating between ligatures and character segment is very hard in open characters. Ligature may be defined as a link between two or more consecutive characters used to join them. In written English language words, two ‘i’ characters side by side may look like ‘u’ and vice versa. Successive ‘n’ and ‘i’ may appear as ‘m’. Character ‘w’ may give the illusion of presence of two characters ‘i’ and ‘u’.

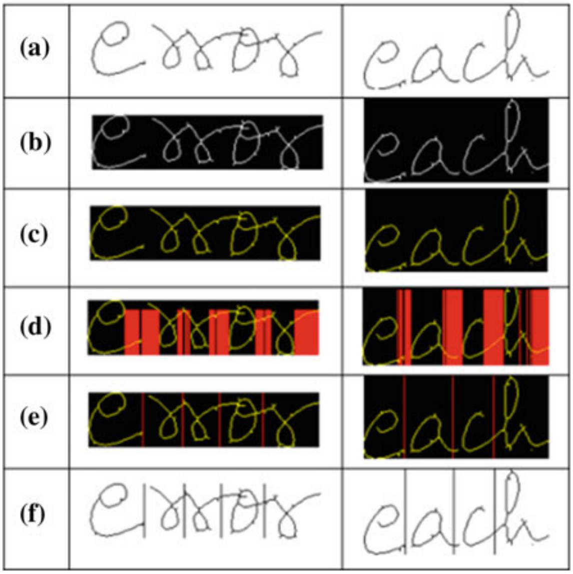


Fig. 8 **a** Images after preprocessing, **b** binary inverted images, **c** images in RGB, **d** over-segmentation, **e** solving over-segmentation problem, **f** output handwritten word images after segmentation

3.2 Methodology

After inverting the handwritten word image, the number of white pixels is counted in each column scanning the image from top to bottom. The columns having 1 or 0 as the count of white foreground pixels are termed as candidate segmentation columns (CSCs), and their positions are noted. Figure 8d shows all such acknowledged columns.

3.3 Problem of Over-Segmentation

Several successive CSCs have been grouped together at various places in the handwritten word image resulting in a situation called 'over-segmentation' and is displayed in Fig. 8d. There are three situations in which this problem of over-segmentation occurs. First, when there is a gap between two successive characters and for each column that lay in this gap, the count of the number of white pixels is 0. Second, when there is a ligature between two characters and the sum of white pixels is 1 for all columns through such ligatures in the whole word image. Finally, when there exist characters such as 'u', 'm', 'n', 'w', which contains loop or semi-loop and the count of all the white foreground pixels for each column which crosses the ligatures-within-characters is also 1. Hence, such types of characters are over-segmented.

3.4 Solving the Over-Segmentation Problem

In the situations, when there is a gap between successive characters, each and every CSC in this gap will have 0 white pixels. By taking mean of all CSCs lying in that gap and merging all the CSCs to a sole column, over-segmentation problem has been solved. In other situations, when ligature-within-character is present (e.g., characters 'u', 'v', 'm', 'w') or a ligature connecting two successive characters, a mean of all those CSCs in a group is calculated which are within a distance below threshold range, and these CSCs are merged to a sole segmentation column.

In horizontal direction, the least gap between successive CSCs is called threshold range, and its value is selected in such a manner that it should be less than the thinnest available character's width such as 'l', 'i'. By repetitive experiments performed many times, threshold's value is selected as '8'. Hence, all the CSCs that are within the 8 pixels range distance from another CSC would be merged into a single segmentation column.

4 Implementing the Proposed Technique

The handwritten word images obtained after various preprocessing steps as shown in Fig. 8a are complemented and taken as input to the segmentation algorithm. By inverting the input black and white images, black pixels form the background and white pixels form the foreground as displayed in Fig. 8b. White pixels have been represented by 1, and it is now easy to count the number of white pixels in each and every vertical column of the binary handwritten word images. Now, this binary image is converted to the RGB color arrangement and is displayed in Fig. 8c. It is convenient if we show CSCs in any color (say red) other than black and white as shown in Fig. 8d. It can be clearly seen that every column, whose total count of white pixels is zero or one, vertically dissects the word image and is termed as a CSC. All CSCs lying within the threshold range of 8 pixels from one another, are fused together to draw a single column representing that particular group of CSCs and is called Segmentation Column and is indicated by the Fig. 8e. Now, the image is then inverted again to get the white background and black foreground for the final segmented handwritten word image as displayed in Fig. 8f.

5 Discussion of Results

A random selection of 400 word images contributed by 10 different writers was used in this experiment. To evaluate the proposed segmentation technique, three types of errors were considered, i.e., number of bad-segmented, over-segmented, and miss-segmented words out of a total of 400 words used in the experiment.

Table 1 Segmentation results

Count of word images used in the experiment	Correct segmented word images (percentage)	Incorrect segmented word images (percentage)	Count of word images and type of error		
			Over-segmented images	Miss-segmented images	Bad-segmented images
400	346 (86.5%)	54 (13.5%)	19	11	40

Table 1 shows that 346 words were segmented correctly, and 54 words were segmented incorrectly. Some incorrectly segmented words were bad-segmented as well as over-segmented and are counted in each type of error category while displaying the results in Table 1. This is why $19 + 11 + 40 \neq 54$.

Comparing the results attained by the proposed segmentation technique with the results of other segmentation techniques developed by other researchers in the literature is not so easy because different researchers presented their segmentation results under different constraints and also they used different types of databases. Some researchers made the assumption that the word images are noise free while some researchers gathered the word image samples from different number of contributors. Although, some authors [5, 6] used popular benchmark databases such as IAM and CEDAR, but they selected different number of handwritten word images from these databases and they even rejected some particular complicated word images from the database as per their personal choices.

6 Conclusion and Future Directions

The proposed technique ensures to dissect each and every possible character boundary by over-segmenting the sample word image enough number of times. Another strategy is also adopted that detects groups of many candidate segmentation points that are lying between any two successive characters and then clubs them into a single segmentation point. Whenever a word image contains untouched characters, accurate segmentation is guaranteed by the proposed technique. It performs very well to dissect ligatures connecting two successive closed characters. This technique sometimes over-segments the open characters because the ligature-within-characters look like ligature connecting two characters. The segmentation accuracy of 86.5% delivered by the proposed segmentation technique is quiet excellent, but the scope of improvement is always there. In future work, there is a need to improve some of the preprocessing techniques, e.g., thinning.

References

1. Tan, J., et al.: A new handwritten character segmentation method based on nonlinear clustering. Neurocomputing **89**, 213–219 (2012)
2. Fujisawa, H.: Forty years of research in character and document recognition-an industrial perspective. Pattern Recogn. **41**, 2435–2446 (2008)
3. Saeed, K., Albakoor, M.: Region growing based segmentation algorithm for typewritten and handwritten text recognition. Appl. Soft Comput. **9**, 608–617 (2009)
4. Choudhary, A., Rishi, R., Ahlawat, A.: A New character segmentation approach for off-line cursive handwritten words. Proc. Comput. Sci. **17**, 88–95 (2013)
5. Marti, U., Bunke, H.: The IAM database: an english sentence database for off-line handwriting recognition. Int. J. Doc. Anal. Recogn. **15**, 65–90 (2002)
6. Hull, J.J.: A database for handwritten text recognition. IEEE Trans. Pattern Anal. Mach. Intell. **16**, 550–554 (1994)

Developing Speech-Based Web Browsers for Visually Impaired Users

Prabhat Verma and Raghuraj Singh

Abstract In this work, we discuss various issues and challenges related to Web Browsing by visually impaired (blind/partially blind) Web users. We also share the design and implementation details of a speech-based Web Browser developed by us for this purpose. The system has been implemented in programming language C# and .NET 4.0 Framework. Microsoft Speech API (SAPI 5.0) has been used for narration of the text and user input feedback. The objective of presenting this work is to promote developments in this neglected area to fill the wide gap between the capabilities of proprietary and freeware screen reader.

Keywords Speech browsers · Web Browsers for visually impaired users · Accessibility · Navigability · Web content management

1 Introduction

This work addresses some of the important issues related to Web accessibility in the context of visually challenged users. Accessibility refers to the ability of a user to use a resource possibly in a different way despite disabilities or impairments. For the Internet-based applications, accessibility implies that a disabled user can navigate, interact, perceive, understand, and contribute to the Web. Speech is a convenient medium of interaction for visually challenged users, and Internet accessibility for them is made possible by providing an alternative speech-based interface for human–computer interaction.

P. Verma (✉) · R. Singh
Computer Science & Engineering Department, Harcourt Butler
Technological Institute, Kanpur, India
e-mail: pvluk@yahoo.com

R. Singh
e-mail: rscse@rediffmail.com

R. Singh
Kamla Nehru Institute of Technology, Sultanpur, India

S. S. Agrawal et al. (eds.), *Speech and Language Processing for Human-Machine Communications*, Advances in Intelligent Systems and Computing 664,
https://doi.org/10.1007/978-981-10-6626-9_12

Problems associated with speech-based Web interfaces are manifold. Most of the Web content available today has been designed for the visual interface via graphical browsers. Visually impaired users may not perceive the layout information and hence the structure of the Webpage. Besides, speech as the medium of interface is essentially sequential in nature. Therefore, it may take a long time for a visually impaired user to go to the point of interest on a Webpage.

Assistive technologies (ATs) such as screen readers make use of underlying Document Object Model (DOM) structure [1] of the Webpage to narrate its contents to a visually challenged user. To ensure that ATs work correctly on a Webpage, Web developers must follow the W3C and other guidelines [2–5] while creating the Web sites. Unfortunately, due to lack of awareness among Web developers, this requirement is not adequately met, and as a result, a large amount of Web content remains inaccessible to ATs and visually challenged users. Web 2.0 has further increased this trend by empowering the end user with Web authoring capabilities. Besides, rich content and dynamic nature of Web 2.0 has also created a lot of trouble in accessibility [6], [7]. The role of ATs is thus to expose such inaccessible Webpage contents using some clever techniques and to present them before the visually challenged user.

Despite the above-mentioned shortcomings of screen readers, these have been the primary tool for using Internet by visually challenged user. Unfortunately, most of the popular and workable screen readers are proprietary and bear a heavy price tag. For example, the cost of JAWS [8], a popular screen reader by Freedom Scientific [9] for single-user license, is around $900. This cost is 10 times higher than that of Windows 7 operating system! The cost is evidently too high to be afforded by an average Indian individual with visual disability. High cost is also attributed to small product market for assistive tools. There are ATs in freeware domain, but they are not popular since most of them may not provide adequate functionalities. Thus, there is a need to design and develop usable as well as affordable assistive tools for visually impaired users.

In this paper, we present the design and implementation of an improved yet affordable and easy-to-use Web Browser for visually challenged users. The system has been implemented in programming language C# and .NET 4.0 Framework. Microsoft Speech API (SAPI 5.0) has been used for narration of the text and user input feedback.

2 Issues and Challenges

In this section, we describe some of the important accessibility-related issues that are prerequisite to the design and development of Web Browser for visually challenged users.

2.1 Keyboard-Based Accessibility

Keys Used in Accessibility by Visually Challenged
To browse Webpages effectively, navigation keys are to be used by the visually challenged users. The navigation keys are: TAB, SHIFT + TAB, CTRL + TAB, CTRL + SHIFT + TAB, UPARROW, DOWNARROW, LEFTARROW, and RIGHTARROW. Besides, PGUP, PGDOWN, HOME, CONTROL, and ALT keys are also used to control the narration of the Webpage.

Keyboard Shortcut-Based Accessibility
Visually impaired users seldom use mouse since it requires coordination of eyes and position of the hand. It is convenient for them to use keyboard shortcuts for inputting commands in GUI environment. Therefore, screen readers provide a set of keyboard shortcuts to use their various functionalities by visually challenged users. They learn these keyboard shortcut-based commands to use the Web effectively. Therefore, it is desirable that all the screen readers follow a standard notation for keyboard shortcut-based commands meant for accessibility.

2.2 Role of Web Developers

Assistive technologies (ATs) such as screen readers make use of underlying Document Object Model(DOM) structure [1] to narrate the Webpage elements to a visually challenged user. To ensure that ATs work correctly on a Webpage, Web developers must follow the W3C and other guidelines while creating the Web sites. Unfortunately, due to lack of awareness among Web developers, this requirement is not adequately met, and as a result, a large amount of Web content remains inaccessible to ATs and visually challenged users. An important checkpoint that a Web developer should perform before launching of the Web site is to navigate through the links and form controls on a page using the keyboard only (e.g., using the "Tab" key) and to make sure that all links and form controls can be accessed without using the mouse and that the links clearly indicate what they lead to.

2.3 Role of Assistive Tools

Various assistive tools for using Web by visually impaired users have been designed using approaches such as context-based approach, semantic approach, annotation-based approach, text summarization. These assistive tools try to enhance the power of visually impaired user by performing one or more of the following changes:

- Provide TTS (text-to-speech) service, i.e., speaking out the content of Webpage and giving speech feedback to user input by echoing the character typed (basic service).

- Make the search informed using some heuristics, thereby reducing the time taken to search some information on Webpage.
- Provide better control over Webpage element by means of shortcut keys.
- Take some otherwise inaccessible content.
- Take some otherwise unreachable link/form element.
- Reduce the number of links (performances) required to traverse to reach to some element on Webpage.
- Simplify the Webpage both in structure and in content.
- Provide a better understanding of Webpage layout/structure.
- Provide a better understanding of images by means of reading out their ALT text.
- Provide a better understanding of visual diagrams by interpreting them.

2.4 Role of Visually Challenged Users

Screen readers are sophisticated programs with a lot of features and functionalities; an average user, whether sighted or visually impaired, is required to use most of the features available with them in order to effectively use them. As compared to visual user interfaces that are self-explanatory to the sighted users, screen readers may use several keyboard shortcuts to access the Web effectively. The usage of these shortcuts is to be learnt by visually challenged users. There may be several modes in which visually challenged user may use screen reader depending on the need for access. Each mode may be activated by a keyboard shortcut to be pressed by user. Thus, proper training and learning of visually challenged users is also a key to effective use of assistive technology.

2.5 Universal Versus Local Installation

An assistive tool may require to be locally installed on user machine, or it may be provided online as a Web service. The first approach constrains the use of the assistive tool by its availability in installed form, whereas in the second approach, local installation is not required. Thus, user can access the Web using any public terminal.

2.6 Challenges Posed by Web 2.0

Web 2.0 is characterized by rich visual contents, user-centric in form and contents. User, who was earlier at receiving end, has become the content provider and Web author. The role of site owner has been reduced merely to managerial and business logic provider. This development has posed a lot of challenges to assistive technology. End users while authoring the Web may not follow the accessibility guidelines due to unawareness. Thus, certain contents provided by them may not be accessible to the assistive technology. Second, dynamic features of Web 2.0 prevent screen readers to correctly access the Web content. The content narrated by the screen reader may completely change during narration and prevent the screen reader to correctly render the contents to the user. Rich contents such as embedded image/button links and anchors remain inaccessible to the screen readers.

3 Design Details

In this section, we describe the design issues, various modes, features, and use cases of the speech-based Web Browsing system.

The proposed system has the following modes of working.

3.1 Link Navigation Mode

Using link navigation mode, visually impaired users can navigate among links to quickly reach to the Webpage to get the desired information. Link navigation mode can help understanding the map of the Web site. Thus, he or she can access the basic structure of the Web site.

3.2 Navigate All Modes

In the second mode, each and every element of the Webpage is traversed by the user in a controlled manner. This mode provides better control for moving around the Webpage and is helpful in reading reports, getting some inaccessible content, etc.

3.3 Newsreader Mode

This mode is useful for e-newspaper reading, magazine reading, or e-learning. In newsreader mode, the contents of the Webpage are narrated automatically by the browser.

3.4 Query Mode

Query mode can be used by the visually challenged user to search the Web site for an input text string. This feature makes use of Google's "Search within the Site" feature.

3.5 Webpage Analytical Mode

This mode speaks out the page statistics of the Webpage, e.g., title, headings, domain, background color, links, image Alt text. This gives the visually challenged user the context of the Webpage very quickly.

3.6 Text Glimpse Through Mouse Mode

This mode may be suitable for users with low vision who may not read out the text correctly but assess the layout information and headings. In this mode, whenever a user right clicks his mouse over a portion of text, the narrator reads out that portion of Webpage.

3.7 Switch Over Among Modes

A visually impaired user may change the browsing mode as per the requirement and suitability through designated shortcut key for that purpose. Thus, through a combination of two or more modes, user can control the navigation and perform the task he or she wishes to perform.

3.8 Keyboard Shortcuts

Keyboard shortcuts are a convenient means to interact with the computer system by the visually impaired users. Using these keyboard shortcuts, a user can perform various activities such as to go to the beginning of Webpage, to go to the next heading, to go to the next form, to go to the address bar. These keyboard shortcuts have to be learnt and remembered by the visually impaired users. A list of such keyboard shortcuts is given in Table 1.

Table 1 Keyboard shortcuts for WACTA

Keyboard command	Usage
Tab	Forward navigate to the next focusable element on the Webpage
Shift + Tab	Backward navigate to the previous focusable element on the Webpage
↓ (Down)	Forward navigate to the next HTML Element on the Webpage
↑ (Up)	Backward navigate to the previous HTML Element on the Webpage
→ (Right)	Increment the speaking rate of the narrator voice
← (Left)	Decrement the speaking rate of the narrator voice
Control	Pause the speaking of the narrator
Escape	Cancel all the speaking of the narrator
Space Bar	Resume the speaking of the narrator
Alt + L	Go to the address (location) bar of Web Browser
Alt + G	Activate the glimpse mode
Alt + N	Activate the news style mode
Alt + A	Activate the analysis mode
Alt + I	Activate the interactive mode
Alt + Q	Activate the query within the Web site mode

3.9 Use-Case Analysis

Use-case diagrams for the WACTA Web Browser are shown in Fig. 1. It depicts various modes and user categories for the usage of the system.

4 Implementation Details

The following subsections describe the implementation details of speech-based Web Browser WACTA for visually challenged users.

4.1 Platform and Language

The WACTA Web Browsing system has been developed for Windows 7 operating system using .NET platform with C# programming language.

In the following subsections, we detail the implementation details through details of classes used in the development.

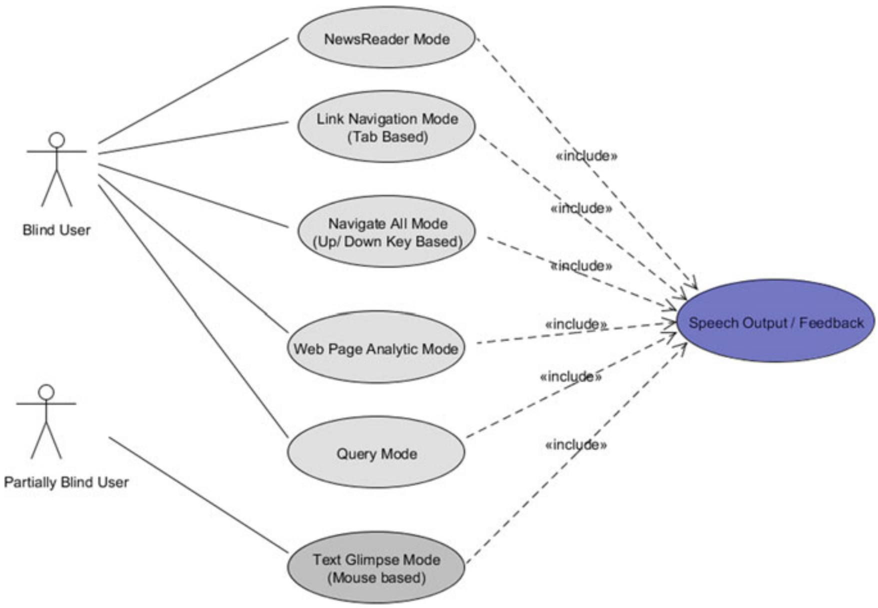


Fig. 1 Use-case diagram of WACTA Web Browser

4.2 *The Web Browser Class*

To create the skeleton of the speech-based Web Browser WACTA, the Web Browser class (Namespace: System.Windows.Forms) has been used. The Web Browser class offers rich features and functionalities that can be used to control the output of various Webpage elements in the desired order.

4.3 *Webpage Analytics*

To get the access to various elements of a Webpage, its DOM structure [1] is explored. Document Object Model (DOM) is an Application Programming Interface (API) for valid HTML and well-formed XML documents. It is based on an object structure that closely resembles the structure of the documents it models. It allows applications to dynamically access content, structure, and style of the documents. DOM is not restricted to a specific platform or programming language.

Following classes have been used in extracting the HTML Elements from Webpages [10], [11]:

HtmlDocument Class This class provides top-level programmatic access to an HTML document hosted by the Web Browser Control. It belongs to System. Windows.Forms namespace.

HTML Element Class HTML Element class represents an HTML Element inside of a Webpage. It belongs to System.Windows.Forms namespace.

HTML Element Collection Class HTML Element Collection class defines a collection of HTML Element objects.

SpeechSynthesizer Class SpeechSynthesizer class of .NET Framework has been used to speech enable the Webpage using the WACTA Web Browser. The class belongs to System.Speech.Synthesis Namespace. SpeechSynthesizer class provides access to the functionality of the installed a speech synthesis engine, i.e., Microsoft Speech API 5.0 (SAPI).

To ensure the consistency in speech output/feedback in various modes of working, only single instance of SpeechSynthesizer class has been created and used throughout the application. This approach prevents the running of multiple narrator instances simultaneously.

4.4 Tab-Based Navigation

The Tab key is the primary mechanism for navigating of a Webpage by visually challenged user. The Tab key visits only those controls with a tab stop. Tab key-based navigation allows to traverse through all the focusable User Interface (UI) elements. These elements have their tabIndex property set to a positive integer value. Tab-based traversal allows user to visit the focusable elements in the increasing order of the tabIndex value of the elements. Normally, HTML links, anchors, and form elements are assigned tabIndex value to enable them to get focus. However, any HTML Element can be enabled by the Web designer to get focus. Tab-based navigation is useful for tasks such as form filling, choosing a Web site from search results, etc.

The code for Tab key-based navigation has been implemented in the handler for “Focusing” event of HTML Element currently getting user input focus, i.e., ActiveElement. The Focusing event is registered in the DocumentCompleted method of the Web Browser class.

4.5 Up/Down Key-Based Navigation

Up/down key-based traversal allows user to visit each HTML Elements of the Webpage in forward/backward order of their creation. This mode has been implemented in the event handler of PreviewKeyDown event of the Web Browser class. Registering of this event is made in the DocumentCompleted method as shown in Table 3.8a. The handler stores each element underneath the BODY Tag in the object of HTML Element Collection type. Problems encountered in this implementation are that InnerText method of HTML Element object returns the text which includes the text of all the children nodes. Thus, it is required to separate the text of the current node only so that it can be spoken out. This issue has been resolved by selecting only those children of HTML Element Collection that has a single child. Still, some elements may not be covered using this method for which string processing is used.

4.6 Narrator Voice Management

The WACTA browser makes use of SpeechSynthesizer class to narrate the Webpage elements in the controlled and desired order. A single instance of this class has been used throughout the system to avoid the multiple-voice output streams which may create confusion. Various methods of SpeechSynthesizer class have been used to enable changes in voice characteristics of narrator such as volume, rate, pitch, gender. These methods have been implemented on the handler of PreviewKeyDown event of Web Browser Class.

4.7 Speech Feedback for User Inputs

For all user inputs such as form filling and Webpage address filling, a provision of speech feedback for each key press is made in the software. It helps in visually impaired user to correct any mistake in typing through listening it.

4.8 Commands and Keyboard Shortcuts in WACTA

Table 1 lists various commands and keyboard shortcuts assigned to the WACTA Web Browser.

5 Results and Discussions

The WACTA Web Browser fulfills the basic requirements of visually challenged users to access the Internet effectively for both routine and important tasks. The design and development of the WACTA Web Browser is a step forward in the direction of providing an effective yet affordable Web accessibility solution for visually challenged users.

Following are the distinct features of the WACTA Web Browser:

- Completely developed using .NET managed code only. Thus, reliability and security are enhanced through automatic garbage collection, automatic bound check, etc.
- Work for both secure and insecure Web sites.
- Work for both direct server- and proxy server-based connections.
- Compatible with Windows 7 operating system.
- Various available browsing modes ensure informed search to quickly locate the required information.
- Specialized speech-enabled browser having range of functionalities which can be easily extended or enhanced.
- Provision of mouse-based accessibility for partially visually challenged users.

WACTA Web Browser has certain limitations as well. First, unlike screen readers such as JAWS which works for Windows Applications as well, WACTA is only a Web Browser. It is meant only for using Internet. Thus, it acts as complimentary to Windows Narrator which comes with Windows operating system. These two can be easily integrated to make a screen reader comparable to JAWS. Besides, WACTA does not work with rich contents such as FLASH. At present, it does not read PDF document, and thus, user may have to download the document and use the inbuilt narrator of PDF Reader software. These issues shall be taken up in future to make the system work for these technologies.

6 Conclusions and Future Plans

The solutions to accessibility problem demands cooperation and coordination among all concerns, e.g., Web site owners, Web designers, authoring tool developers, assistive technology developers, researchers, public terminal (computer) facilitators, policy makers, visually challenged users. Awareness regarding accessibility issues may bring these parties to work in close coordination.

Assistive technology needs continuous and concerted efforts to keep pace with the fast developments in the Web technology. The problem is, till the time, ATs find solution to a current Web issue, an altogether new issue is already set. Traditionally, an industry criterion for making investment on a project has been determined by the number of use cases offered by the aimed product. Thus, investments on ATs may not gain due importance due to its small market size.

This research work has strengthened our belief that knowledge and technology should be used in favor of mankind to every possible extent. Internet is a wonderful tool having ability to compensate the visual impairment with its technology. Design and development of powerful yet affordable speech-based interfaces would be certainly helpful in enhancing the overall quality of life of visually challenged.

The work done by us shall be further taken up in future, e.g., enhancing the features and capabilities of WACTA Web Browser, its design and development for Android-based tablet computers as well as extending it for Hindi-scripted Web sites.

References

1. Document Object Model [Internet], http://www.w3.org/DOM/[^]. Access July 2014
2. Web Accessibility Initiative (WAI) [Internet] [updated 2010], The World Wide Web Consortium (W3C). http://www.w3.org/WAI/
3. User Agent Accessibility Guidelines (UAAG) Overview [Internet] [updated 2009], The World Wide Web Consortium. http://www.w3.org/WAI/intro/uaag.html
4. Authoring Tool Accessibility Guidelines (ATAG) Overview [Internet] [updated 2008], The World Wide Web Consortium. http://www.w3.org/WAI/intro/atag.php
5. Evaluating Web Sites for Accessibility: Overview [Internet] [updated 2009], The World Wide Web Consortium. http://www.w3.org/WAI/eval/Overview.html
6. Cooper, M.: Accessibility of emerging rich web technologies: web 2.0 and the semantic web. W4A2007- Keynote (2007)
7. WAI-ARIA Overview [Internet] [updated 2009]. The World Wide Web Consortium. http://www.w3.org/WAI/intro/aria.php
8. http://www.practicalecommerce.com/articles/2114-Screen-Readers-Eight-Frequently-Asked-Questions [Practical E-Commerce]. Access July 2015
9. Freedom Scientific. http://www.freedomscientific.com/. Access on June 2012
10. MSDN Online Developer Resource
11. Schildt H.: The Complete Reference C# 4.0. Tata McGraw Hill (2010)

Adaptive Infrared Images Enhancement Using Fuzzy-Based Concepts

S. Rajkumar, Praneet Dutta and Advait Trivedi

Abstract Image enhancement is the process of modifying digital images so that results are suitable for human perception. An upcoming need for image visualization during all lighting conditions by the use of infrared (IR) imagery has gained momentum. It is deemed fit for efficient target acquisition and object deduction. However, due to low image resolution and difficulty in spotting certain objects whose temperature is similar to that of the ground, infrared images must be subjected to further enhancement. Our given proposal aims to enhance infrared images, making use of the fuzzy-based enhancement technique (FBE), and to compare its efficacy with other techniques such as histogram equalization (HE), adaptive histogram equalization (AHE), max–median filter, and multi-scale top-hat transform. The enhanced image is then analyzed using different quantitative metrics such as peak signal-to-noise ratio (PSNR), image quality index (IQI), and structural similarity (SSIM) for performance evaluation. From experimental results, it is concluded that FBE results in the best quality image.

Keywords Infrared images · Histogram equalization · Adaptive histogram equalization · Fuzzy sets · Fuzzy enhancement

S. Rajkumar (✉) · A. Trivedi
School of Computer Science and Engineering, VIT University, Vellore 632014, India
e-mail: rajkumars@vit.ac.in

A. Trivedi
e-mail: advait.trivedi2012@vit.ac.in

P. Dutta
School of Electronics Engineering, VIT University, Vellore 632014, India
e-mail: praneet.dutta2012@vit.ac.in

S. S. Agrawal et al. (eds.), *Speech and Language Processing for Human-Machine Communications*, Advances in Intelligent Systems and Computing 664,
https://doi.org/10.1007/978-981-10-6626-9_13

1 Introduction

The ability of infrared (IR) imaging to generate detailed and coherent images under varied conditions of light exposure makes it useful for applications such as night vision, thermography, hyper-spectral imaging, tracking, heating, climatology, astronomy, biological systems, photo bio-modulation. Its application is also extended to the Military Domain where short-, long-, and mid-wave IR imaging is used in target tagging [1]. However, image quality is generally prone to degradation due to image capturing devices or environmental conditions. Thus, effective capture of high quality picture can often be cumbersome. Use of the degraded infrared image for further processing such as image retrieval, image segmentation, feature extraction, and object detection makes derivation of expected result difficult. This problem is what necessitates efficient image enhancement. There are several conventional enhancement techniques adopted for this purpose.

Histogram equalization [2] is considered as one of the most common image enhancement techniques because of its relative simplicity in implementation. The method focuses on the distribution of pixels along the graphical histogram thus providing an enhancement of the contrast. However, this methodology fails to enhance images which are attributed by physically distant gray pixel population. The shortcomings of the traditional histogram equalization led to the development of a technique which could locally make contrast enhancement to distant clusters of gray pixels. Thus, adaptive histogram equalization performs a locally varying input–output image enhancement. However, the adaptive method aggravates the existing noise in the imagery or may even lead to distortion of pixels. In Bi-Histogram Equalization (BBHE) [3] technique the existing graphical histogram is split into two sub-histograms. This splitting is done by intelligently selecting partitioning points on the histogram. The method then goes on to treat each sub-histogram as a separate entity and normalizes them exclusively. Thus, brightness of the image is retained as compared to the HE technique. However, this is applicable only for brightness preservation in images to avoid superfluous information. Minimum Mean Brightness Error Bi-Histogram Equalization (MMBEBHE) [4], a developed strategy for BBHE, can be embraced for higher degrees of protection. The detachment depends on limit level, which would yield least Absolute Mean Brightness Error (AMBE). The objective behind this strategy is to give the most extreme level of brightness preservation in BBHE to detail a proficient, recursive and whole number-based answer for rough the yield mean as a component of the threshold level. In any case, MMBEBHE now and then shows poor brightness preservation and upgrade, and especially in pictures that require significantly more brightness preservation, and it neglects to control the over improvement of the picture. Range Limited Bi-Histogram Equalization (RLBHE) [5] has been proposed, which separates the input histogram into two free sub-histograms by an edge that limits the intra-class change. This was completed to successfully isolate the articles from the background. It accomplishes an outwardly all the more satisfying complexity improvement while keeping up the input brightness and is anything but difficult to execute in real time. A variant of the self-adaptive contrast enhancement [6] algorithm is the plateau-based technique. Threshold value is adopted, consisting of the local minima present in the histogram of the image. However, it is more suitable for continuous probability histograms. By the addition of a relevant lower and upper threshold value to a double plateau is another method proposed. Upper threshold is primarily useful for the suppression of the background and the lower threshold for the enhancement of the contrast of the image. However, they may overlap and result in poor enhancement. An approach proposed by Lin [7] for IR image enhancement derived from an adaptive and high-boost filter. This method works best when the initial threshold is calibrated to the mean of the image. However, groups of pixels tend to dominate the histogram when the objects are miniature when juxtaposed with the background. Liang et al. [8] described an adaptive algorithm—upper and lower threshold values of the double plateau are computed at runtime based established on gray level distribution.

Furthermore, non-histogram-based enhancement methods in different domains such as filtering, wavelet transforms, morphological operators are reviewed. Filtering methods [9] fail to accomplish effective enhancement when substantial background noise encompasses the target. Wavelet transform methods [10] fail to detect the non-dominant features of the object. Morphological operators' method [11] is inefficient when clutter is heavy and target is dim. To overcome these problems, fuzzy-based enhancement is proposed.

The advantage of the proposed method is that there is no need of prior information about IR image; the required parameters are adaptively modeled from the input image. The result of the proposed method shows a better visualization than the existing methods, which is witnessed in Sect. 3.

The rest of the paper is organized as follows: In Sect. 2, the proposed methodology of our given method is described. In Sect. 3, the experimental results and performance analysis are elucidated in depth and the Sect. 4, concludes the work.

2 Proposed Methodology

The block diagram proposed method shown in Fig. 1.

2.1 Histogram Equalization

The primary motive in mind while applying histogram equalization is to increase the global contrast of gray level pixels, thereby flattening the resulting histogram [2]. The resultant picture will have a wider dynamic range of pixel. The main steps

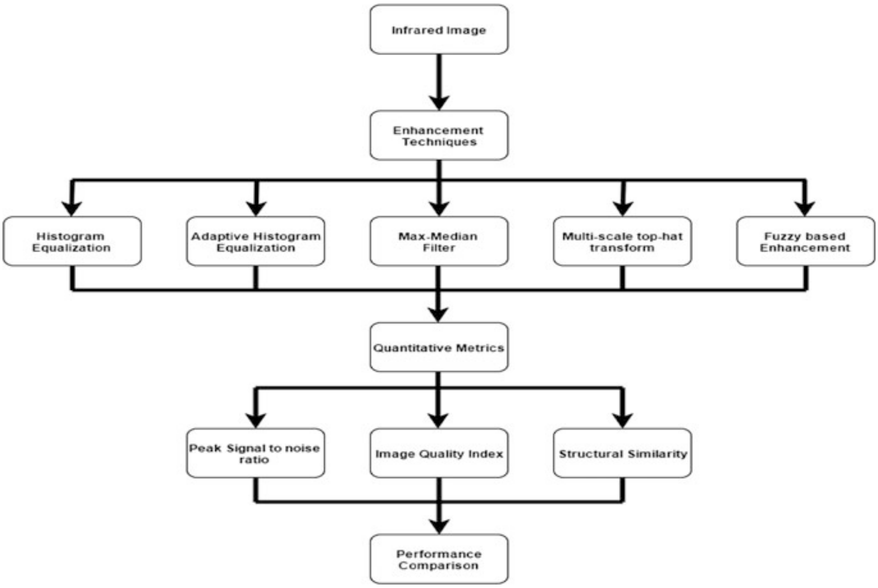


Fig. 1 Block diagram of proposed method

involved in the equalization process are a translation of pixels from one histogram representation scheme to another more contrast stretched one. The translation is done over a cumulative distribution function.

2.2 Adaptive Histogram Equalization

Adaptive histogram [12] aims at making local changes to the pixel contrast in a picture. The inspiration behind adaptive histogram was to overcome the inhomogeneous pixel densities that stand out from the general contrast of the image even after applying traditional histogram. Thus, a scheme to analyze the neighborhood of pixels, while making the equalization transformation was adopted. Adaptive histogram defines the size of the neighborhood around the translating pixels. Thus, the resulting pixel transformation is directly correlated with the neighboring pixel echelon resulting in contrast enhancement of small-scale pixel clusters while contrast diminishes of large-scale pixel clusters.

2.3 Max–Median Filter

The max–median filter is a sequential enhancement technique where the input image matrix first undergoes median filter transformation. In this step, the 3 × 3 filter window is selected and traversed through the input image vector. The second step involves the max filter transformation. The advantage of this step is the enshrinement of fine features which might have been distorted in the first process [9].

2.4 Multi-scale Top-Hat Transform

Multi-scale top-hat transformation [11, 13] uses mathematical set operations such as erosion (Θ) and dilation ($\oplus$) to bring about a transformation. During this method, two entities, namely the image vector F and its structuring element (mask) S, are taken as inputs. The first morphological operation, dilation, outputs the maximum value of the pixels in the pixels' vicinity. Erosion does just the opposite as it outputs the minimum value of the pixel in the pixels' vicinity. Other morphological operations such as opening ($\circ$) and closing ($\cdot$) use sequential use of the above two elementary set operations. The multi-scale top-hat transform uses these morphological tools mentioned above in contrast enhancement of the input image matrix.

2.5 Fuzzy-Based Enhancement

The IR image [14] may become obscured as a result of atmospheric conditions—light and heavy rain, heavy haze, cloudy skies as well as sensor difficulties (lens deviation, optical defocusing, etc.) can play a major part in this. These cases can also take place concurrently and make the image subject's pose and shape unclear. Fuzzy-based enhancement technique provides a solution to mitigate this.

Fuzzification. Fuzzification is the first step of fuzzy image processing. It consists of converting the image from spatial domain into the fuzzy domain. It can be defined as:

$$\mu_{ij} = T(x_{ij}) = \left[1 + \frac{x_{\max} - x_{ij}}{F_{\mathrm{d}}}\right]^{-F_{\mathrm{e}}} \tag{1}$$

Here $x_{\max}$ is the maximum intensity level in the given input image; F_{e} and F_{d} denote the exponential and denominational fuzzifiers. When $x_{\max} = x_{ij}$ then, $\mu_{ij} = 1$ indicating the maximum brightness.

Fuzzifiers. F_{d} which is the denomination fuzzifier is calculated using Eq. (2), and the computed exponential fuzzifier F_{e} is to be the mean of the image.

$$F_{\mathrm{d}} = \frac{x_{\max} - x_{\min}}{\left(\frac{1}{2}\right)^{-1/x_{\min}} - 1} \tag{2}$$

Contrast Intensification Operator (INT). Contrast intensification operator (INT) is used to modify the fuzzified image. The μ$_{ij}$ membership value is modified by the INT and is shown in Eq. (3) as:

$$\mu_{ij}^{'} = \begin{cases} 2\left[\mu_{ij}\right]^{2}, & 0 < = \mu_{ij} < = 0.5 \\ 1 - 2\left[1 - \mu_{ij}\right]^{2}, & 0.5 < \mu_{ij} < = 1 \end{cases} \tag{3}$$

Defuzzification. The fuzzy image, thus reshaped is converted back into spatial domain using inverse transformation. This is called defuzzification as shown in Eq. (4)

$$E(i,j) = T^{-1}\left(\mu_{ij}^{'}\right) = x_{\max} - F_{\mathrm{d}} * \left(\left(\mu_{ij}^{'}\right)^{-1/F_{\mathrm{e}}}\right) + F_{\mathrm{d}} \tag{4}$$

2.6 *Quantitative Measures*

Quantitative assessments performed on the improved images with the help of different quantitative measures help in extracting and analyzing necessary data from the images. The accompanying segment clarifies the quantitative measures utilized as a part of the investigation of the enhancement strategies.

Peak Signal-to-Noise Ratio (PSNR). This measure quantifies the enhanced quality of the transformed image to that of the original one [15]. Its mathematical definition is as follows:

$$\mathrm{PSNR} = 10\log_{10}\left(\mathrm{MAX}^{2}/\mathrm{MSE}\right) \tag{5}$$

$$\mathrm{MSE} = \frac{1}{pq}\sum_{i=0}^{p-1}\sum_{j=0}^{q-1}\left[E(i,j) - I(i,j)\right]^{2} \tag{6}$$

where *MAX* represents the maximum value in an image. p, q are the height and weight of an image. $I(i,\ j)$ is the value of the input image, and $E(i,\ j)$ is the value of enhanced image.

Image Quality Index (IQI). The definition in Eq. (7) represents image distortion in terms of factors such as correlation loss, luminance variance/distortion, and contrast distortion [16]:

$$\mathrm{IQI} = \frac{((4 * \sigma_{\mathrm{IE}})(\mu_{\mathrm{I}} + \mu_{\mathrm{E}}))}{(\mu_{\mathrm{E}}^2 + \mu_{\mathrm{I}}^2) * (\sigma_{\mathrm{E}}^2 + \sigma_{\mathrm{I}}^2)} \tag{7}$$

where σ_{IE}—Covariance of IE, μ_{I}—Average of I, μ_{E}—Average of E, σ_{I}^2—variance of I, and σ_{E}^2—variance of E. The range of IQI value is between −1 and 1. If I and E are similar, then the IQI value is 1 and −1 which indicates the dissimilarity.

Structural Similarity Index (SSI). It measures the similarity between the two images [17, 18]. SSIM is an improved version of the peak signal-to-noise ratio. It is defined as:

$$\mathrm{SSIM} = \frac{((2\mu_{\mathrm{E}}\mu_{\mathrm{I}} + C_1) * (2\sigma_{\mathrm{EI}} + C_2))}{((\mu_{\mathrm{E}}^2 + \mu_{\mathrm{I}}^2 + C_1) * (\sigma_{\mathrm{E}}^2 + \sigma_{\mathrm{I}}^2 + C_2))} \tag{8}$$

where μ_{E} and μ_{I} denote the average intensities of image E and I, σ_{E} and σ_{I} denote the variance of image E and I, σ_{EI} gives the covariance of E and I, C_1 and C_2 are constants. The value of SSIM index varies from −1 to 1. The value 1 indicates the identical between two images.

3 Experimental Results and Performance Analysis

The experimental result of the proposed method is tested with an OTCBVS IR image database. The presented results in this paper are derived from the OSU thermal pedestrian database. The database contains 10 classes of pedestrian images. These classes have 284 images with 984 pedestrian objects each of size 360 × 240 and have been captured in different environmental conditions such as heavy rain, light rain, haze and cloudy. These conditions impede accurate object detection. Thus, necessitating the need for further infrared image enhancement. The sample image along with the transformed image output fitting the proposed enhancement technique can be seen in Fig. 2.

Figure 3 shows the subjective comparison of the results achieved from HE [2], AHE [12], max–median filter [9], multi-scale top-hat transform [11], and FBE. It can be inferred from the result of the above study that FBE outperforms other image enhancement methods considered. An empirical conclusion of the above inference can be drawn by observing from Tables 1 and 2. In both tables the numeric value recorded for the given quantitative measure, FBE shows higher values as compared to other techniques. Similarly, Fig. 4 graph indicates that the FBE method values close to 1 compared to others for IQI measure.

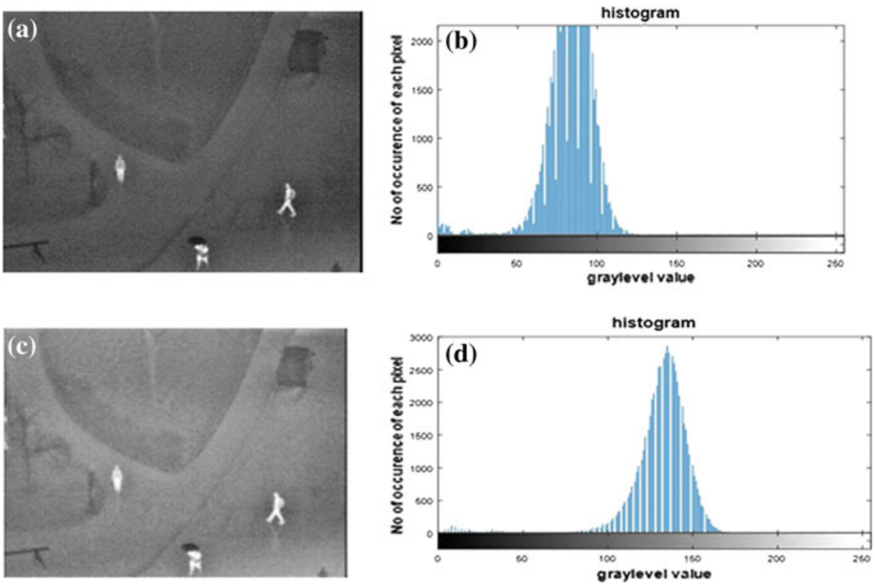


Fig. 2 Sample input and output image with corresponding histogram. **a** Input image. **b** Histogram of (**a**). **c** The result of proposed method. **d** Histogram of (**c**)

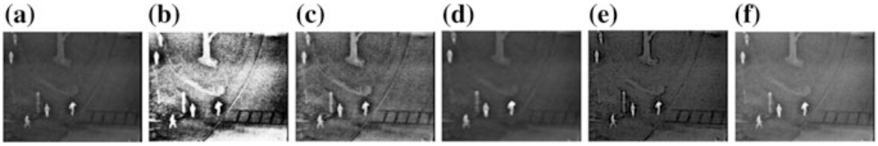


Fig. 3 Subjective comparison of the enhancement results. **a** Input image. **b** The result of HE. **c** The result of AHE. **d** The result of max–median. **e** The result of Multi-scale Top-hat Transform. **f** The result of proposed method

4 Conclusion

This paper enumerates the need for IR fuzzy-based enhancement. A comparative study encompassing the focused method with traditional image enhancement techniques using quantitative measures to evaluate performance was done. A detailed numeric comparison highlights the proficiency of the proposed method over other conventional methods.

Table 1 Comparative analysis of PSNR values

Classes	Methods				
	HE	AHE	Max–median	Multi-scale top-hat	Fuzzy-based enhancement
1	10.23404	18.9154	29.26478	27.10886	15.17621
2	10.27299	18.85632	29.26035	27.04929	14.74572
3	10.29752	18.93346	29.21745	27.05025	14.9137
4	10.26339	18.88489	29.24569	27.09574	14.95716
5	10.32499	19.00813	29.19873	27.02321	15.01931
6	10.2423	18.95163	29.29329	27.14411	14.97015
7	10.22546	18.87212	29.2925	27.23615	14.92502
8	10.33082	18.97642	29.22492	27.20112	14.98895
9	10.2473	18.7605	29.30436	27.18122	15.11788
10	10.27534	18.8299	29.25023	27.15497	15.15694

Table 2 Comparative analysis of SSIM values

Classes	Methods				
	HE	AHE	Max–median	Multi-scale top-hat	Fuzzy-based enhancement
1	0.265415	0.612153	0.792895	0.828057	0.90953
2	0.269712	0.612157	0.791431	0.824699	0.901216
3	0.268671	0.612124	0.791861	0.828797	0.906003
4	0.266866	0.612449	0.792132	0.830017	0.905576
5	0.270274	0.613125	0.791948	0.831725	0.908231
6	0.271531	0.611602	0.792565	0.830797	0.907536
7	0.275485	0.611745	0.791231	0.821126	0.888375
8	0.274659	0.611934	0.791118	0.821634	0.8888531
9	0.274544	0.613248	0.793386	0.812567	0.893067
10	0.27122	0.613176	0.792329	0.821634	0.896448

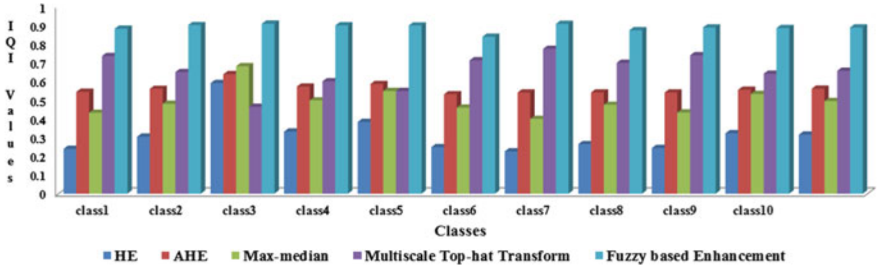


Fig. 4 Comparison of IQI values with HE, AHE, max–median, multi-scale top-hat transform

References

1. Rajkumar, S., Chandra Mouli, P.V.S.S.R.: Target detection in infrared images using block-based approach. In: Informatics and Communication Technologies for Societal Development, pp. 9–16. Springer India (2015)
2. Gonzalez, R.C.: Digital Image Processing. Pearson Education India (2009)
3. Kim, Y.-T.: Contrast enhancement using brightness preserving bi-histogram equalization. IEEE Trans. Consum. Electron. **43**(1), 1–8 (1997)
4. Chen, S.-D., Ramli, A.R.: Minimum mean brightness error bi-histogram equalization in contrast enhancement. IEEE Trans. Consum. Electron. **49**(4), 1310–1319 (2003)
5. Zuo, C., Chen, Q., Sui, X.: Range limited bi-histogram equalization for image contrast enhancement. Opt. Int. J. Light Electron Opt. **124**(5), 425–431 (2013)
6. Wang, B., et al.: A real-time contrast enhancement algorithm for infrared images based on plateau histogram. Infrared Phys. Technol. 48(1), 77–82 (2006)
7. Lin, C.-L.: An approach to adaptive infrared image enhancement for long-range surveillance. Infrared Phys. Technol. **54**(2), 84–91 (2011)
8. Liang, K., et al.: A new adaptive contrast enhancement algorithm for infrared images based on double plateaus histogram equalization. Infrared Phys. Technol. **55**(4), 309–315 (2012)
9. Deshpande, S.D., et al.: Max-mean and max-median filters for detection of small targets. In: SPIE's International Symposium on Optical Science, Engineering, and Instrumentation. International Society for Optics and Photonics (1999)
10. Zhao, J., Qu, S.: The fuzzy nonlinear enhancement algorithm of infrared image based on curvelet transform. Proc. Eng. **15**, 3754–3758 (2011)
11. Bai, X., Zhou, F., Xue, B.: Infrared image enhancement through contrast enhancement by using multiscale new top-hat transform. Infrared Phys. Technol. **54**(2), 61–69 (2011)
12. Pizer, S.M., et al.: Adaptive histogram equalization and its variations. Comput. Vis. Graph. Image Process. **39**(3), 355–368 (1987)
13. Serra, J. Image Analysis and Mathematical Morphology. Academic Press, Inc. (1983)
14. Soundrapandiyan, R., Chandra Mouli, P.V.S.S.R.: Perceptual Visualization Enhancement of Infrared Images Using Fuzzy Sets. Transactions on Computational Science XXV, pp. 3–19. Springer, Berlin (2015)
15. Sayood, K.: Introduction to data compression. Newnes (2012)
16. Wang, Z., Bovik, A.C.: A universal image quality index. IEEE Signal Process. Lett. **9**(3), 81–84 (2002)
17. Wang, Z., et al.: Image quality assessment: from error visibility to structural similarity. IEEE Trans. Image Process. **13**(4), 600–612 (2004)
18. Lewis, J.P.: Fast normalized cross-correlation. In: Vision Interface, vol. 10, no. 1 (1995)

Toward Machine Translation Linguistic Issues of Indian Sign Language

Vivek Kumar Verma and Sumit Srivastava

Abstract Sign language is a gesture-based language for communication of deaf people. It is basically a nonverbal symbolic language which is usually used to speak with and by deaf people. Indian sign language is a linguistically under-investigated language. Research on Indian sign language linguistics is also limited because of unavailability of standard sign dictionary and the unavailability of such tools which provide any education for Indian sign language. In the interpretation between sign language and verbal spoken language, there is an intermediate step in which the sign language needs to be represented by some written notation. However, there is as yet no standard notation for Indian sign language. In this paper, we investigate intermediate notation form for sign language as suitable for the machine translation. We also discuss the challenges of other linguistic issues for machine translation on Indian sign language.

Keywords Indian sign language · Machine translation · Sign notations

1 Introduction

Machine translation is a field of natural language processing which manages computerized interpretation. It is the procedure by which a software program is utilized to translate content from one common dialect, (for example, English) to another, (for example, Hindi). Automated translation of spoken languages to Indian sign languages has received much attention over the last decade [1–3]. Sign linguistics, then again, is a moderately new field of study in the region of linguistics, and it combines machine interpretation of communications via gestures.

V. K. Verma (✉) · S. Srivastava
Department of Computer Science and Engineering, Manipal University, Jaipur, India
e-mail: vivekkumar.verma@jaipur.manipal.edu

S. Srivastava
e-mail: sumit.srivastava@jaipur.manipal.edu

S. S. Agrawal et al. (eds.), *Speech and Language Processing for Human-Machine Communications*, Advances in Intelligent Systems and Computing 664,
https://doi.org/10.1007/978-981-10-6626-9_14

Gesture-based communications as dialects have not been mulled over as broadly as talked dialects, and there is still much to be found out about them.

2 Challenges in Indian Sign Language Translation

Visual sign-based communications are not just interpreted type of spoken language, as they are frequently seen to be, yet are done trademark lingos with complex phonetic segments [4]. The unlucky deficiency of an adjusted relationship between a stamped vernacular and a spoken lingual is one of the challenges went up against by sign-to-text or text-to-sign structures, in light of the way that it infers that there are fundamentally two stages to the strategy: generation and understanding [5]. Natural language tools limit within the same vernacular, however, between two media (voice and substance); lingo translation programming, while making a mapping beginning with one tongue then onto the following, generally limits within the same medium (i.e., substance to message). Correspondence through marking generation would need to fuse both generation and understanding, working between two media and furthermore two dialect as the sign dialect and the spoken dialect [6, 7].

The real issues to be considered in a machine interpretation framework from a source as spoken dialect to a target as gesture-based communication accompanying are:

- Developer should have a strong understanding of source language as well as target language.
- Developer should have a strong understanding of the grammar for both source language and target language.
- Developer should have a strong understanding of the syllables and dialects for both source language and target language.
- Developer should have a strong understanding of the social conventions, culture, user, and expectations for both source language and target language.

The phonetic data that is accessible about the gesture based communication, the semantic analysis of communications, through signing is not almost as complete with respect to their spoken counterparts, and numerous communications via gestures the world over have not been phonetically researched [8, 9]. The linguistic information that is available for Indian sign language is a major issue. The linguistic analysis of sign languages is not nearly as complete as for spoken language has, and most of the sign languages around the world have not been linguistically investigated as should for the research. The decision to select appropriate parser for the source language (natural language like; English) impacts the semantic data that is accessible for the creation of the communication through signing; specifically, discourse analysis is regularly expected to design with all possible signs [10–12]. Indian sign language generation needed exact mapping from spoken language for interpretation of a configuration and suitable graphical animation.

3 Machine Translation Approaches for Sign Language

Machine translation is an automated process by which one natural language can be translating into another natural language. In case if one of the natural language is sign language needs special attention to process. Generally, the process of machine translation can be functionally classified as direct machine translation, rule-based machine translation, and corpus-based translation [13, 14].

As in direct translation process, the source natural language is analyzed according to the structure using morphology level and then it is modeled for the target language. The execution of a direct machine translation framework relies on upon the quality and amount of the source-target dialect lexicons, morphological investigation, content handling tools, and word-by-word interpretation with minor linguistic alterations on word request and morphology.

The most favored methodology for machine translation is corpus-based machine translation. In this approach, a bilingual content corpus is taken and prepared to get the expected structure. Generally, corpus-based methodology is utilized under two categories, statistical approaches and the example-based approach for machine translation. As in case of statistical approach, the dialect model decides the probability of the target dialect for picking the right word in the interpreted dialect. The interpretation model then again serves to register the restrictive probability of the target dialect given the source dialect for the most part indicated with probability of target and source language [15]. In the final stage, the greatest probability of result of both the dialect model and the interpretation model is processed which gives the measurably in all probability plausible sentence in terms of target dialect.

Example-based translation approach normally utilizes past interpretation cases to make an interpretation in terms of source to target dialect. The fundamental concept behind of example-based translation approach is to recover samples of existing interpretation in its case base and give the new interpretation taking into account of that case. In this approach, most part happens in three stages as matching, alignment, and recombination [16, 17].

4 Sign Notations

Communication via gestures is represented to outwardly, and it cannot be read and write as other composed spoken language. There are few research attempts to compose communication via gestures; however, these attempts are not usable on account of their limitations [18]. Indian sign language contains gestures that are hard to comprehend and learn. Some of the approaches are discussed here to compose sign language with the end goal of utilizing it as a part of machine interpretation. A notation framework for gesture-based communication is firmly expected to propel the investigation of its structure. In spite of the fact that further study is expected to make the sign writing system less difficult, more justifiable and absolutely all inclusive, the automated translated system is by all accounts ready to add to the improvement of a deaf community and individuals [19].

Gesture-based communication for hard hearing individuals has special components that are truly unique in relation to those of normal spoken people. Communication through signing is a famous dialect contrasted with talked dialect, which is a greater amount of a subjective one. Another issue needs to be in contrast that between the two languages for machine translation if one is sign language it does not have its own writing system. In this way, so as to compose depictions of signs, line drawings, photos, and delineations have generally been utilized; however, these speak to just a little minute as compared to actual visual signing process. This way of representation for visual sign is called notations. A notation representation system for gesture-based communication was firmly expected to advancement the investigation of its structure.

Stokoe notation [20] is utilized as a part of word references, a few etymology books, and research articles having it in Unicode will be of awesome advantage for the research community. Another important notation system HamNoSys is proposed by IDGS, University of Hamburg on the basis of German sign language [21]. This notation has been designed on the concept of physical actions taken by signer while communications and is not based on the meaning of sign. Each notation of the HamNoSys indicates physical actions like hand shape, hand orientation, hand position, motion. As in Fig. 1, a sign of numeral three depicts in both Stokoe and HamNoSys notations.

One more notation system popularly used for representation of gestures is Gloss Notations [22]. Gloss-based notation depends on identification of individual sign. Indian sign languages are typically translated with this notation as word-by-word using English gloss. Prosody can be also glossed as superscript words and the corresponding scope under brackets. As if we have an English sentence "Man is not Deaf" and for prosody as "not" use a negation term in the superscript indicated below.

$$\text{“Man is not deaf”} > \text{MAN IS [DEAF]}^{\text{NEGATIVE}}.$$

The HamNoSys is a phonetically based documentation framework at first written by hand, yet a machine clear textual style is accessible from the University of Hamburg.

A few points of interest of HamNoSys notation are that it is based on any particular features of sign as mentioned in Table 1. It has a versatile framework

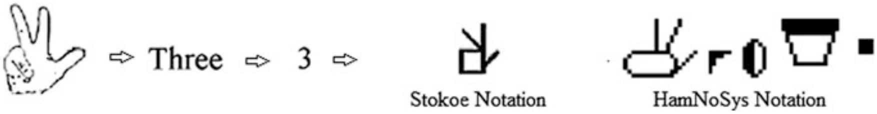


Fig. 1 Notations for sign of numeral three

Table 1 HamNoSys features

Features	Hand shapes	Hand orientation	Location
Composed	Basic forms and diacritics for thumb position and bending	Hands and the corresponding direction	Natural distance of the hand from the body
Components involve	The fingers involved or the form of individual fingers	Finger direction specify degrees of freedom, and palm orientation determining the third degree	First component the location within the plane, second is distance of the hand from the body
Size (set of instruction)	Large	Small	Small

with formal language structure and can be put away in a database. Then again, it does not give any simple approach to represent nonmanual features of sign, for example, outward appearances.

Another important notation specified for sign language is sign writing (SW) to annotate gestures in sign languages. Furthermore, the notation was developed not only for one language but was built to be appropriate for any sign language. A vast number of sign languages are already making use of the script. The goal of sign writing is to enable signers to be literate in their first language, not requiring them to learn another language in order to read and write. SW is a pictorial notation system and can describe nonmanual features [23]. It makes use of a set of symbols that can be combined to describe any sign. Though standardization, efforts are being put into place, and the system is still flexible; if a language cannot describe a sign with the available symbols, it is possible to add to the set [24].

5 Challenges in Sign Translation

In another spoken language translation, all the words in one language may have corresponding words in another language, but in case of sign there are no words. In some cases of sign translation word in one language is to be expressed by group of signs or vice versa. The given languages may have completely different structures with sign language, for example, English has “Sub-Verb-Object” structure, while Indian sign language has generally “Sub-Object-Verb” structure. For machine translation there is need to map parts of speech but in sign translation lack of one-to-one correspondence of parts of speech. The ways in which sentences are put together also differ among different languages. A word can have more than one meaning, and sometimes group of words or whole sentence may have more than one meaning in a language. Words ambiguity is the major issue with almost all the language in the world. All the translation problems cannot be solved by applying corresponding grammar, especially when there is no such grammar available for sign language. Translation requires not only vocabulary and grammar but also knowledge from past experience. The developer should have a strong understanding of the rules under which complex human language operates and how the mechanism of this operation can be simulated by automatic means for sign translation. The simulation of sign language behavior by automatic means is hard to achieve as the language is open and dynamic system in constant change.

6 Conclusion

The steps in translation of sign languages have scope for much improvement. This work has focused on the intermediary step between Indian sign recognition and spoken language synthesis, analyzing sign language notation systems and how they can be used to optimize the translation process. Taking everything into account, we might want to fortify the note worthiness of deaf studies and the significance of sign education in India. Beginning with an overview of sign linguistics discussion of the special challenges for machine translation under Indian environment. Additionally, we tried to describe briefly the different existing approaches that have been used to develop machine translation systems for different regions. Through the review process we conclude that all the machine translation of Indian spoken language uses statistical and hybrid approaches. We have applied all the discussed notations with different conditions in ISL and recommended sign writing is best suited. Sign writing provided a good notation for translation, though the other systems have not been completely ruled out and may still be successfully used in translation systems. It has a machine readable format which is continuously being improved, and data in sign writing is easily available. A corpus of sign writing data can be constructed, cleaned, and translated using a translation tool for sign.

References

1. Verma, V.K., Srivastava, S., Kumar, N.: A comprehensive review on automation of Indian sign language. In: 2015 International Conference on Advances in Computer Engineering and Applications (ICACEA). IEEE, New York (2015)
2. Kishore, P.V.V., Rajesh Kumar, P.: A video based Indian sign language recognition system (INSLR) using wavelet transform and fuzzy logic. IACSIT Int. J. Eng. Technol. **4**(5) (2012)
3. Cooper, H., Holt, B., Bowden, R.: Sign language recognition. In: Visual Analysis of Humans. Springer, London (2011)
4. Sandler, W.: The phonological organization of sign languages. Lang. Linguist. Compass **6**(3), 162–182 (2012)
5. Dzikovska, M., et al.: BEETLE II: deep natural language understanding and automatic feedback generation for intelligent tutoring in basic electricity and electronics. Int. J. Artif. Intell. Edu. **24**(3), 284–332 (2014)
6. Stein, D., Schmidt, C., Ney, H.: Analysis, preparation, and optimization of statistical sign language machine translation. Mach. Transl. **26**(4), 325–357 (2012)

7. Curiel, A., Collet, C.: Sign language lexical recognition with propositional dynamic logic. arXiv preprint arXiv: PP.1403.6636 (2014)
8. Sako, S., Kitamura, T.: Subunit modeling for Japanese sign language recognition based on phonetically depend multi-stream hidden markov models. In: Universal Access in Human-Computer Interaction. Design Methods, Tools, and Interaction Techniques for Einclusion, pp. 548–555. Springer, Berlin (2013)
9. Caridakis, G., Asteriadis, S., Karpouzis, K.: Non-manual cues in automatic sign language recognition. Pers. Ubiquit. Comput. **18**(1), 37–46 (2014)
10. Borghi, A.M., et al.: The body and the fading away of abstract concepts and words: a sign language analysis. Front. Psychol. **5** (2014)
11. Friginal, E., et al.: Linguistic characteristics of AAC discourse in the workplace. Discourse Stud. (2013)
12. Chiu, M.M.: Statistical discourse analysis of an online discussion: cognition and social metacognition. In: Productive Multivocality in the Analysis of Group Interactions, pp. 417–433. Springer US (2013)
13. El Kholy, A., Habash, N.: Orthographic and morphological processing for English–Arabic statistical machine translation. Mach. Transl. **26(**1–2), 25–45 (2012
14. Kaliszyk, C., Urban, J.: Learning-assisted automated reasoning with Flyspeck. J. Autom. Reason. **53**(2), 173–213 (2014)
15. Cho, K., et al.: Learning phrase representations using RNN encoder-decoder for statistical machine translation. arXiv preprint arXiv: 1406.1078 (2014)
16. Sall, S.B., Sharma, R., Shedamkar, R.R.: Example based machine translation using natural language processing. Int. J. Sci. Eng. Res. **4**(8), 1771–1776 (2013)
17. Dong, T., Cremers, A.B.: A novel machine translation method for learning Chinese as a foreign language. In: Computational Linguistics and Intelligent Text Processing, pp. 343–354. Springer, Berlin (2014)
18. Zafrulla, Z., et al.: American sign language recognition with the kinect. Proceedings of the 13th International Conference on Multimodal Interfaces. ACM, New York (2011)
19. Guimarães, C., et al.: Deaf Culture and sign language writing system—a database for a new approach to writing system recognition technology. In: 2014 47th Hawaii International Conference on System Sciences (HICSS). IEEE, New York (2014)
20. Hutchinson, J.: Literature Review: Analysis of Sign Language Notations for Parsing in Machine Translation of SASL (2012)
21. Kaur, R., Kumar, P.: HamNoSys generation system for sign language. In: 2014 International Conference on Advances in Computing, Communications and Informatics (ICACCI). IEEE, New York (2014)
22. Porta, J., et al.: A rule-based translation from written Spanish to Spanish Sign Language glosses. Comput. Speech Lang .**28**(3), 788–811 (2014)
23. Bouzid, Y., et al.: Towards a 3d Signing Avatar from Signwriting Notation. Springer, Berlin (2012)
24. Bouzid, Y., Jemni, M.: A virtual signer to interpret signwriting. In: Computers Helping People with Special Needs, pp. 458–465. Springer International Publishing (2014)

Analysis of Emotion Recognition System for Telugu Using Prosodic and Formant Features

Kasiprasad Mannepalli, Panyam Narahari Sastry and Maloji Suman

Abstract Speech processing has emerged as one among the most important application areas of digital signal processing. In the present world, the speech processing has become essential for technological developments in various aspects and this technology is also incorporated in many gadgets. Emotion-based recognition is where the emotion of the person is identified from the differences in stress and other properties of speech. The features such as intensity, formants, bandwidth, and pitch vary with the change in emotion. These changes are identified, and the emotion is recognized with respect to the average value in that particular emotion. This project aims to recognize the emotion in Telugu language. The speeches of different speakers are collected for the same sentence in three different emotions (happy, neutral, and bore), and various features are extracted from these collected speeches. Finally, an algorithm is proposed to recognize the emotion based on the features extracted. Its applications are access control, transaction authentication, law enforcement, etc. The recognition accuracy to recognize the emotion of the speaker is 79%.

Keywords Speech processing · Emotion recognition · Telugu
Prosodic features · Formant features

1 Introduction

Speech is an important part of human interaction. Hence, a lot of research in this area is being done and is an open area of research. The different research areas within speech processing, like machine learning, speech recognition, are the most sought after among the researchers in the last few decades. Speech recognition is the research domain in which the meaning of the speech is extracted. There are

K. Mannepalli (✉) · M. Suman
K L University, Vijayawada, Guntur (Dist), Andhra Pradesh, India
e-mail: mkasiprasad@gmail.com

P. N. Sastry
CBIT, Hyderabad, Telangana, India

S. S. Agrawal et al. (eds.), *Speech and Language Processing for Human-Machine Communications*, Advances in Intelligent Systems and Computing 664,
https://doi.org/10.1007/978-981-10-6626-9_15

various emotions that a person can exhibit. The detection of these changes in emotions is done in emotion recognition. This difference in emotions exists due to the psychological behavior of the person. The detection of these emotions is necessary and is used in criminal detections, truth detector machines, etc. An important challenge for the research in the area of speech processing and technology is the understanding and modeling of individual variations in spoken language. Individuals have their own style of speaking, dialect, accent as well as their social background. This work focuses on automatic emotion identification of a speaker, given a Telugu speech sample which can be employed to improve the speech recognition system.

2 Literature Review

Hansen and Varadarajan [1] have presented their work on speech production in the noisy environment resulting in the Lombard effect. This effect has a serious impact on the efficiency of speech systems. The Lombard speech was produced using varied levels of noise and was analyzed by using features like duration patterns, energy histograms, and spectral tilt. Gaussian mixture model (GMM) was used as classifier. The average EERs were improved, and EERs for matched and unmatched adaptation and testing conditions were obtained as 4.75 and 12.37%, respectively. EER was as low as 1.78% was obtained at the highest noise level by adapting neutral speaker models.

Andreas Stolcke, Sachin S. Kajarekar, Luciana Ferrer, and Elizabeth Shrinberg have described [2] a novel approach for speaker recognition. They have used the maximum likelihood linear regression (MLLR) adaptation transforms as features and support vector machine (SVM) for classification. They have shown that how MLLR–SVM approach can be enhanced by the combination of transforms relative to multiple reference models. A comparison is made between two techniques for compensating for intersession variability (WCCN and NAP) as applied to MLLR–SVM system. The results obtained have shown that the NAP is highly sensitive to the choice of data for procuring covariance statistics, and the WCCN is very much influenced by the choice of background set.

Thomas, Eriksson et al. [3] have studied the selection of features for speaker recognition from the information theory view. They have reported that the classification error probability to the mutual information across the speaker's identity and features is closely related. Qualitative statements about feature selection can be made by using information theory. Features like different LPC parameterizations and mel-warped cepstral coefficients were studied.

Ortega-Garcia et al. [4] have proposed a method on speaker recognition in the area of security applications through speech input. Nevertheless, variability of speech degrades the performance of speaker recognition. The external variability and intra-speaker variability sources produce mismatch across training and testing phases. The channel and intersession variability would be explored for accomplishment of real automatic speech systems for the commercial as well as forensic speaker recognition. The experiments have shown that combination of score normalization and CMN techniques reduce ERR significantly.

Khalid Saeed and Mohammad Kheir Nammous in their work "A Speech-and-Speaker Identification System: Feature Extraction, Description, and Classification of Speech-Signal Image" [5] explained a speech-and-speaker (SAS) identification system based on recognition of spoken Arabic digits. The speech samples of numbers from zero to ten collected in Arabic language. The conventional and the neural network-based classifications were used. The successful recognition of the speaker identifying system touched about 98.8% in few cases. The average overall successful recognition is about 97.45% in recognition of the uttered word and identifying its speaker. For a three-digit password, the percentage of accuracy obtained is 92.5%.

Aronowitz and Burstein [6] presented the efficient techniques for speaker recognition. These techniques involve approximated cross entropy (ACE) for approximation of the Gaussian mixture modeling (GMM) likelihood scoring. The training and testing session was represented by Gaussian mixture modeling. The algorithm was efficient in performing speaker recognition with very less degradation when compared to classical Gaussian mixture model algorithm.

Besson et al. [7] proposed an algorithm that explores a theoretical framework for the extraction of optimized audio capabilities using video information. A simple way of assessing the mutual information (MI) between the audio and video characteristics as a result which allows the identification of the active speaker. This approach performs the optimization of an objective function based on the MI. While estimating of the probability density function (pdf) of characteristics and the cost function, approximation is not needed. This algorithm achieved a speaker identification rate of 100% on in-house test sequences, and for most commonly used sequences, the obtained efficiency is 85%.

Parthasarathi et al. [8] presented a paper on "Privacy-Sensitive Audio Features for Speech/Non-speech Detection." The objective of the work was to examine the features of speech and non-speech detection (SND) having low linguistic information. Three different approaches were examined for privacy-sensitive features. Methods for instantaneous feature extraction, excitation source information, and feature obfuscation such as local (less than 130 ms) temporal averaging. Randomization is applied on the information of excitation source. The application of obfuscation methods on the excitation features resulted in low phoneme efficiencies in concurrence with SND performance comparable to that of MFPLP.

Dharanipragada et al. [9] have presented a robust technique of feature extraction for continuous speech recognition. The important aspect of the technique is the minimum variance distortionless response (MVDR) method of spectrum estimation. They have incorporated perceptual information in two approaches: (1) after the MVDR power spectrum is computed and (2) directly during the MVDR spectrum estimation. The technique used was MVDR spectral envelope estimation. The proposed feature extraction method gave a lower WER compared to mel-frequency cepstral coefficients (MFCC) and perceptual linear prediction (PLP) feature extraction in number of cases. The technique is most robust to noise though the performance in clean conditions is not degraded.

Wu et al. [10] presented a paper on “Robust Multifactor Speech Feature Extraction Based on Gabor Analysis.” Robust sparse features extraction is done using a multifactor analysis method by the data sample processing in tensor structure. From the results, it may be stated that this method can improve the speech recognition system performance, particularly in noise conditions when compared to the traditional methods of speech feature extraction.

3 Features

3.1 Prosodic Features

Prosodic features are suprasegmental in nature. Prosodic features are not confined to a particular segment. They occur at upper levels of a speech utterance. The prosodic features vary with language and are the undeniable phonetic spurts or chunks of speech. The prosodic units are marked by phonetic cues which include the characteristics of prosody like pitch, accents, duration, and intonation. Generally, pitch can change from sentence to sentence because of rising and falling intonations. Prosody helps in resolving sentence ambiguity. Prosodic cues such as pauses and intonation change will make the meaning of the word clear for the louder speech samples. Prosodic feature is robust. Prosodic information is not only language-specific; speaker-specific prosody is also available.

3.2 Pitch

Pitch is a perception feature that is shown in frequency scale, and it presents the ordering of sounds on a frequency-related scale. Pitch is compared as higher and lower when associated with musical melodies. This requires sound with clear and stable frequency to distinguish from noise. Pitch may be measured as a frequency and varies from person to person. The pitch varies even if the emotion of the speech varies. In this work, pitch calculated using autocorrelation was used.

3.3 Pitch Accent

Pitch accent is a characteristic of some languages, so that the hue variations can be used to distinguish words, but where potentially distinct tones are restricted to only one or two syllables of a word—as opposed to fully tonal languages as the standard

Chinese with each syllable can have an independent voice. In a tone language accent, timbre phonemic syllable is typically one that is acoustically prominent.

3.4 Intensity

Intensity is a feature that measures stress on the word or syllable level of the speech signal. Sound intensity can be defined as the sound power per unit area. The noise in the sound is measured at the receivers end as sound energy quantity.

3.5 Formants

Formants are the characteristics that are non-uniform spectrum adaptation signal that is in resonance frequencies of the vocal tract and usually have higher signal-to-noise ratios than the other parts. Formants are important features that distinguish the words. The order of these frequencies along the frequency axis may be different, depending on the phonemes and position of the window along the phoneme (i.e., at the starting or ending of the phoneme). Apart from the formant, i.e., resonant frequency, bandwidth, and magnitude spectrum can be used in the special frequency to encode the properties of speech. These properties can be used in different applications like voice recognition, speech enhancement, noise removal, and adaptive filters. In this work, the first four formants F0, F1, F2, and F3 are used as the formant features.

4 Methodology

4.1 Methodology

Eight people were identified for this work, and a sentence given for recording the speech of the speakers was “entraa ila vachhav (Telugu).” Twenty-five speech samples were recorded from each speaker using the phone in three different emotions (happy, bore, and neutral). These files which were initially in .mp3 format were converted into .wav format using media.io. The prosodic features like minimum, maximum, range, standard deviation, and mean absolute slope of pitch, intensity were extracted successfully. The energy, first four formants, and their bandwidths were also extracted. The datasheet for 600 samples was prepared. An average matrix of feature values of all training samples is formed. The individual emotion was represented in a column. The testing is performed using nearest neighborhood classifier by finding the minimum distance between training and testing samples. The test emotion speech is identified as the emotion class with which it has the minimum distance. The efficiency of the system is calculated from the procedure discussed above. This is shown in Fig. 1.

4.2 Speech Samples

India is a nation which can be called as a subcontinent because of diverse culture, number of languages, and different religions. Indians speak different dialects and languages. Telugu is a Dravidian category language, which is spoken by the people who are natives of Telangana and Andhra Pradesh. As standard database is not available for Telugu language, speech database has been created by collecting speech samples from eight different people. The sentence has been recorded as “entraa ila vachhav” in Telugu. Each person speaks the same sentence in three different emotions. Each emotion has been iterated for 25 times. The utterance of the sentence is recorded using HTC smartphone. These speech samples help us to know the speech properties like pitch, frequency, bandwidth, which help us in further distinguishing the emotion from other samples.

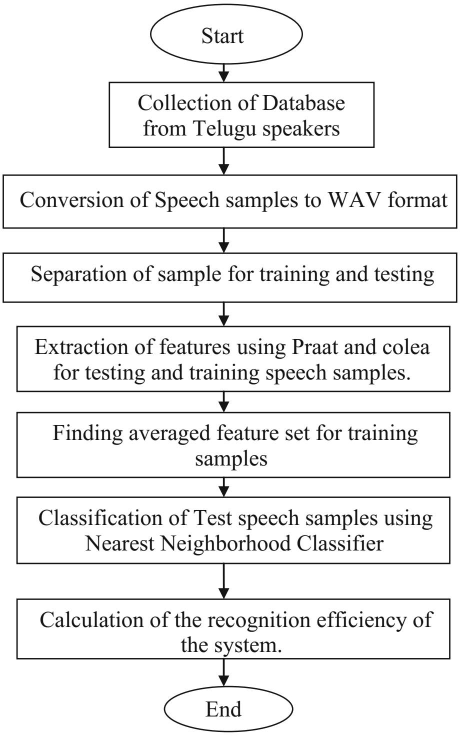

Start
Collection of Database from Telugu speakers
Conversion of Speech samples to WAV format
Separation of sample for training and testing
Extraction of features using Praat and colea for testing and training speech samples.
Finding averaged feature set for training samples
Classification of Test speech samples using Nearest Neighborhood Classifier
Calculation of the recognition efficiency of the system.
End

Fig. 1 Methodology of Telugu emotion recognition system

5 Results

One-twenty samples were used in testing phase for identification. Table 1 shows the results obtained from all samples. It shows 120 test samples and the emotion identification of them among the eight speakers in three different emotions (neutral, happy, and bore). Among the 120 samples from eight speakers in three different emotions, 96 were identified correctly, making the overall efficiency of 80%. Out of 40 samples in neutral emotion, 33 were identified correctly making the efficiency of it as 82.5%. Out of 40 samples in happy emotion, 34 were identified correctly making its efficiency 85%. Out of 40 samples in bore emotion, 29 were identified correctly making the efficiency of it as 73%. This is shown in Table 1.

6 Conclusions

(1) Telugu emotion speech database was developed as there is no standard database for Indian languages as per the literature survey.
(2) A total of 160 samples for each emotion were developed as training database. Further, the testing database of 40 samples for each emotion was successfully developed.
(3) The different emotions collected were sad, neutral, bore from eight speakers and samples for each emotion.
(4) Prosodic features like minimum, maximum, range, standard deviation and mean absolute slope of pitch, intensity were extracted successfully.
(5) Features like energy, first four formants and their bandwidths also were successfully extracted.
(6) The efficiency of emotion recognition system by using the above features and nearest neighborhood classifier (NNC) is found to be 79% on an average.

Table 1 Result of Telugu emotion recognition system

Emotions	Number of Samples	Correctly identified Samples	Percentage of accuracy
Neutral	40	33	82.5
Happy	40	34	85
Bore	40	28	70

7 Future Scope

(1) More number of emotions may be collected for the Telugu emotion speech database.
(2) The emotion recognition accuracy may be increased by using MFCC features.
(3) This recognition accuracy can be improved by using deep neural networks for classification.

References

1. Hansen, J.H.L., Varadarajan, V.: Analysis and compensation of Lombard speech across noise type and levels with application to in-set/out-of-set speaker recognition. IEEE Trans. Audio Speech Lang. Process. **17**(2) (Feb 2009)
2. Stolcke, A., Kajarekar, S.S., Ferrer, L., Shrinberg, E.: Speaker recognition with session variability normalization based on MLLR adaptation transforms. IEEE Trans. Audio Speech Lang. Process. **15**(7) (Sept 2007)
3. Eriksson, T., Kim, S., Kang, H.-G., Lee, C.: An information-theoretic perspective on feature selection in speaker recognition. IEEE Signal Process. Lett. **12**(7) (July 2005)
4. Ortega-Garcia, J., Gonzslez-Rodriguez, J., Cruz-Llanas, S.: Speech variability in automatic speaker recognition systems for commercial and forensic purposes. IEEE AES Syst. Mag. (Nov 2000)
5. Saeed, K., Nammous, M.K.: A speech-and-speaker identification system: feature extraction, description, and classification of speech-signal image. IEEE Trans. Ind. Electron. **54**(2), 887–897 (Apr 2007)
6. Aronowitz, H., Burshtein, D.: Efficient speaker recognition using approximated cross entropy (ACE). IEEE Trans. Audio Speech Lang. Process. **15**(7), 2033–2043 (2007)
7. Besson, P., et al.: Extraction of audio features specific to speech production for multimodal speaker detection. IEEE Trans. Multimed. 10(1), 63–73 (Jan 2008)
8. Parthasarathi, S.H.K., et al.: Privacy-sensitive audio features for speech/non-speech detection. IEEE Trans. Audio Speech Lang. Process. **19**(8), 2538–2551 (Nov 2011)
9. Dharanipragada, S., et al.: Robust feature extraction for continuous speech recognition using the MVDR spectrum estimation method. IEEE Trans. Audio Speech Lang. Process. **15**(1), 224–234 (Jan 2007)
10. Wu, Q., et al.: Robust multifactor speech feature extraction based on Gabor analysis. IEEE Trans. Audio Speech Lang. Process. **19**(4), 927–936 (May 2011)

Simple Term Filtering for Location-Based Tweets Classification

Saurabh Kr. Srivastava, Rachit Gupta and Sandeep Kr. Singh

Abstract Twitter micro-blog is producing a massive amount of data. Here, conversation spreads rapidly among people having same interests and gets exchanged at an amazing speed. **People share their experiences and opinions openly in Twitter on various topics** such as relations, professional work, daily activities, health issues, social issues, food activities. So aligning, reasoning and forecasting the event impact have become the utterly important topics in the related research community. In this way, Twitter platform provides a new pathway for text-related knowledge insights. We conducted a content analysis of two locations' (Arizona and London) Twitter data during day timings to identify people's eating habit and their related discussions on Twitter. According to people's eating habits, we try to identify the eating-related potentials using n-gram analysis, especially for pizza. We have collected 3214-weekend tweets which are collected in 13.33 h. We have evaluated the result using the n-gram analysis to identify positive, negative and neutral sentiments related to pizza food. In the reported result, we evaluated the use of micro-blogging message content as an indicator of opening new eating-related businesses in a particular geographical region.

Keywords Tweets · n-grams · Micro-blogs · Pizza · Food

S. Kr. Srivastava (✉) · S. Kr. Singh
Department of CSE, JIIT University, Noida, India
e-mail: phd.jiit@gmail.com

S. Kr. Singh
e-mail: sandeepk.singh@jiit.ac.in

R. Gupta
Department of Information Technology, ABES-EC, Ghaziabad, India
e-mail: rachit.gupta4@gmail.com

S. S. Agrawal et al. (eds.), *Speech and Language Processing for Human-Machine Communications*, Advances in Intelligent Systems and Computing 664,
https://doi.org/10.1007/978-981-10-6626-9_16

1 Introduction and Related Work

Micro-blogging Websites are creating a new way for enrolling people from various communities that show different cultural as well as moral values. With the advancement of mobile technology, the use of these micro-bloggers is frequent and common. People share their personal and habitual information frequently with the smart phones. LinkedIn, Facebook, Myspace, Twitter, Google+ are some popular micro-blogs. Micro-blog platforms are creating a new pathway for data acquisition and knowledge representation. Micro-blog shares several common features such as creating of public profile, defining the list of other users (friends and followers) whom they share their connection, and viewing and discovering the connection between other users. Micro-blog supports unstructured data, so it is always part of research to identify useful patterns from the unstructured content. People usually communicate with others with their personal feelings and languages, because they communicate in real-time scenario and the content of micro-blogs is used for real-time analysis. User tweets are available to all his/her followers. According to the statistics given in [1], a rough estimate of 54.5 million people are using Twitter in USA with 63% of users under the age of 35 and 45% between 18 and 34. In addition, USA accounts for 51% of all users worldwide. Twitter4J API provides a rich interface to work with Twitter. Twitter has been used to understand the real behavioural and habitual trends. Abdulkareem Alsudais et al. in his paper [1] represented his work to identify the tweets location of the users. In this, six types of location categories are identified for representation with tweets, namely active life, eating out, hotels, nightlife, shopping and shows. Their result can be beneficial for research and business. Fu et al. [2] present a model for trend prediction in micro-blog community. In his study, SinaWeibo micro-blog is used for trend prediction and representation of node vector semantics, and semantics of tweets is understood. Distance between tweets and nodes is also studied. Achrekar et al. [3] have done the work to predict the flu trend. They studied Social Network-Enabled Flu Trend (SNEFT) framework, in which tweets mentioning is used as an indicator of predicting emergence and spread of influenza in people. Alsultanny [4] has done his research work on forecasting the labour market. To predict the needs of labour market, data mining approach has been presented by using three techniques, namely implementing Naive Bayes Classifiers, Decision Trees, and Decision Rules, and comparisons are done between them. Kannan et al. [5] used a data mining approach to forecast the financial stock marketing. They have proposed a technology from which historic data can be used to find hidden patterns that can help investors in investment decisions. Twitter has been used to automatically track flu and cancer activities in [6]. Influenza and cancer activity levels in various regions are tracked and mapped in real time, and output is visualized in the form of interactive maps, pie charts, and time series graphs. Another study in [7] has modelled and discovered public health topics and tobacco. Hirose et al. [8] have searched a prediction strategy which can predict the future trend of influenza by using Twitter data. They combine Twitter data and CDC's influenza-like illness (ILI) data and apply regression model on that data. The main focus of this paper is to find more accurate way of prediction, and they found that instead of single linear regression model, multiple linear regression models with ridge regularization give better and more accurate result. Wang Hao et al. saw that micro-blogs currently effect social communication, their taste, their lifestyle and their thinking [9]. In this paper, they try to identify the reason for the spread of tweets and its behaviour. They have used Susceptible-Infectious-Susceptible (SIS) model for their result, which was developed in medical field that identifies the reason for the spread of infectious disease. The main focus of the model is to predict future retweet trend. Akshay Java et al. observed the micro-blogging trend by analysing topological and geographical data by Twitter; they found that because of a lot of people using micro-blogs by the community, it is easy to find intentions of the community; they also focus on how user with similar intentions connects with each other [10].

Contribution: In this, we have collected tweets based on geographical location parameter (latitude, longitude) [11, 12]. These tweets are extracted based on the keywords' search that is basically related to food-related habits such as pizza, burger, KFC. Tweets are collected during daytime of weekend of two popular cities (Arizona and London). Then, the n-gram analysis is used for classifying tweets into positive, negative and neutral categories. Graphical measure (pie chart) is used for ease of understanding. So this classification of tweets can help us to recommend what type of business will flourish at any geographical location. The rest of this paper is organized as follows: Sects. 2, 3 and 4 present details of evaluated results, processing and discussions. In Section 5, conclusion of the work is explained, and the paper ends with Sect. 6 that explains the future work directions (Fig. 1).

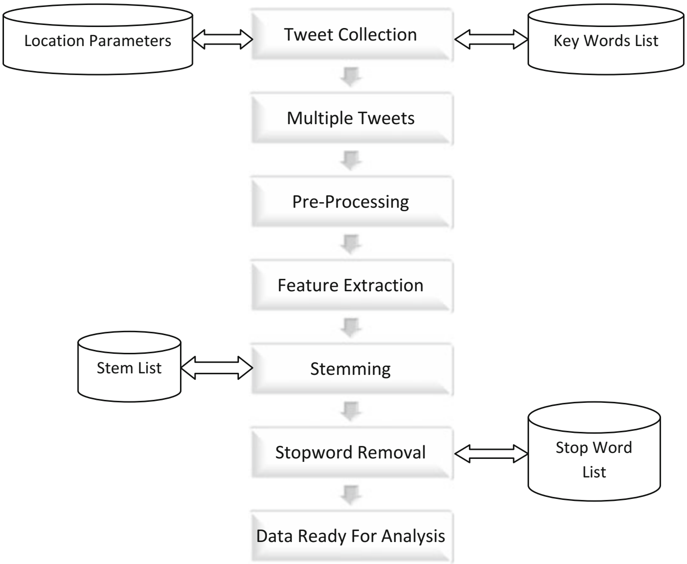


Fig. 1 Pre-processing task

2 Evaluated Results

(1) Data preparation

In order to obtain the tweets to identify the eating habits, we have chosen two popular cities (London and Arizona), and with the use of Twitter API (Twitter4J), we gathered tweets from their latitude and longitude: with London having the latitude and longitude of 51.5072°N, 0.1275°W and Arizona having the latitude and longitude of 34.0000°N, 112.0000°W, within the range of 2 miles from coordinates. The tweets are related to food activities and are extracted with the keywords—pizza, KFC, Domino's, fast food, pizza party, hot dog, snack, McDonalds, food and burger. The data we have collected are usually loosely organized. Unstructured text may lead to poor text analysis that affects the accuracy of an output. So to bring the data into an understandable format, we need to pre-process it. Sample of tweets that we have collected is given in Table 1.

These collected tweets are irrational at this point of time. These tweets we have collected for around 13.33 h during weekend and collection times for tweets in both the cities are same. We have removed irrelevant data using some stopping words. The list of stopping words is applied to the raw data. Java program is used for removing stop words. The list of stopping words is given in Table 2.

To perform the n-gram analysis, stemming is used to make the data in an understandable format. To normalize the sample data, we have used stemming.

Stemming—Stemming reduces words to its root form. To find out the stem/root of a word, the stemming method is used. For example, the words such as materially, materialized, materiality and materialize all can be stemmed to the word 'material'. The main purpose of this process is to reduce the number of words, to remove various suffixes, to save memory space and time and to have exactly matching stems. **Stemming** process is based either on linguistic dictionaries or on various algorithms. The goal of stemming is often removal of derivational affixes. For our stemming process, we have used Snowball Stemmers. Snowball is a language in which stemming algorithms can be easily represented. We have used Snowball Stemmer class for stemming. Figure 2 shows stemming process.

Table 1 Tweets sample

Tweet No.	Tweet's content
T1	These refs treating Gasol likes he's sacred treasure pizza
T2	Off at home at 2:50 what's going on!? The relegation party should have a step longer! #UTC
T3	The Lauriston pizza in the park?? @ Victoria Park https://t.co/t0p9IlN7f0
T4	It's inevitable that whenever lucy is hungover I will receive a snapchat of her dominoes pizza

Table 2 List of stopping words

Stopping words
be, we, so, am, is, are, the, that, for, in, by, and, @, #, $, &, to, above, after, along, e.g., ex, go, it, is, then, was, were and than

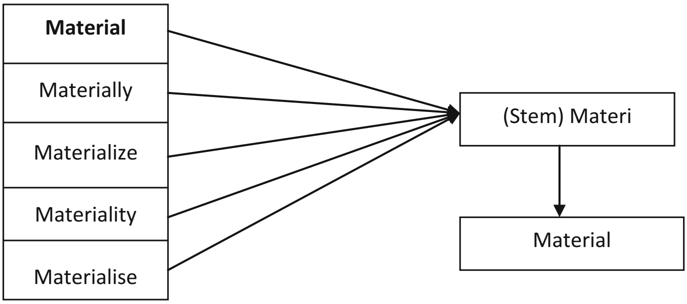


Fig. 2 Stemming process

(2) Data processing (n-gram analysis)

Data processing involves extracting the required information from a given data set.

For processing our pre-processed rational data, we have used Python. Python is a programming language. It is an object-oriented programming language with lots of features which do support in the development of Web applications as well.

The n-gram is basically a set of co-occurring words that are in a given window, and during its computation, we typically move one word forward. These models are extensively used in text mining and natural language processing. The n-gram model is imagined just like placing a small window over a text. The simplest n-gram model is, therefore, the so-called unigram model. In a unigram model, we only look one word at a time.

For example, for the sentence ‘I really like pizza it’s pretty good’, if $N = 2$ (known as bi-gram), then n-grams would be:

- I really
- Really like
- Like pizza
- Pizza it’s
- It’s pretty
- Pretty good

So we have six bi-grams in this case. Note that we moved from I → really to really → like to like, etc., essentially moving one word forward to generate the next bi-gram.

If $N = 3$(known as trigram), the n-grams would be:

Table 3 Sample of collected bi-grams

Bi-grams	Frequency
(‘good’, ‘food’)	8
(‘healthy’, ‘food’)	8
(‘Need’, ‘food’)	6
(‘got’, ‘food’)	6
(‘with’, ‘pizza’)	4
(‘yummy’, ‘foodie’)	4
(‘Burger’, like)	4
(‘dominos’, ‘pizza’)	4
(‘pizza’, ‘with’)	4
(‘love’, ‘cheese’)	4

- I really like
- Really like pizza
- Like pizza it’s
- Pizza it’s pretty
- It’s pretty good

So we have five n-grams in this case. Similarly, if $N = 1$, then this would be the case of unigram. Now for text categorization, we compute bi-grams for each word and then counted its frequency with positive sense word followed by food-related keywords. For the computation of the n-gram analysis, we have used open-source Natural Language Toolkit (NLTK). NLTK is a leading platform for building Python programs to work with human language data. Sample bi-grams are listed in Table 3.

3 Results

There are total 3214 tweets collected from London and Arizona in 13.33 h. in a weekend. These collected tweets are based on the keyword related to food activity. We have consolidated 2037 tweets from London and 1177 tweets from Arizona. Once these tweets are collected, they are pre-processed and final data processing i.e. n-grams are done. From tweets that are collected from London, we are able to construct a total of 15,251 bi-grams, and similarly from Arizona, a total of 11,458 bi-grams are constructed. From these bi-grams, we have identified the positive, negative and neutral sentimental bi-grams, as a prefix followed by food-related keywords.

To identify positive, negative and neutral bi-grams, we have used the keywords given in Table 4.

The outcome of the number of tweets at two cities is given in Table 5.

Table 4 Sentiment keywords

Sentiment	Keywords
Positive	affection, appreciation, fondness, lust, amity, delight, like, love, delight, with, yummy, delicious, amazing, enjoy, zest, healthy and excellent
Negative	dislike, hate, junk, worst, nah, cold, shittest and dirty
Neutral	of, me, my, Gre, ice, service, use, after, bank, festival, kg, e.g., from, desert, competition, English, oven, looks, free, fast, tastes, follow, slice, cooked, honest, actual, about, vegen

Table 5 Tweets data

Location	Positive bi-grams	Negative bi-grams	Neutral bi-grams	Total bi-grams
London	147	23	262	15,251
Arizona	165	11	173	11,458

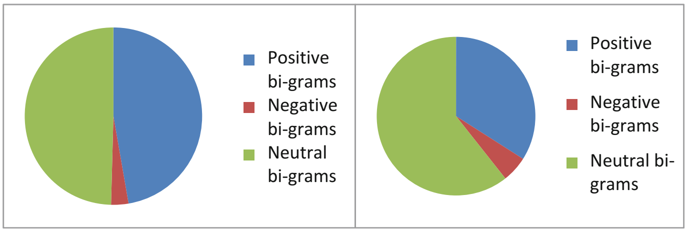


Fig. 3 Pie chart for showing sentiment tweets

4 Discussion

With the analysis of tweets, we found 147 and 165 positive sensed tweets from London and Arizona. These tweets are collected within same time span which is of 13.33 h. The graphical representations of these two are represented in the following pie charts (Fig. 3).

From the pie chart, it is clearly visible that numbers of positive tweets in Arizona are much more than London, during the same time span.

5 Conclusion

We directly address the problem of identifying the potential location, which is vulnerable to food activities. We focus on the tweets which are having food-related keywords such as pizza, burger, KFC. We performed bi-gram analysis and explored the positively sensed bi-grams and concluded our work by classifying the bi-grams into three major categories, i.e. positive, negative and neutral. Based on the results of collected tweets, we can easily depict that Arizona is having more potential for food-related businesses than London. Pie chart is used as a graphical measure to make the understanding more clear.

6 Future Work

The work can be further extended for analysis to get better results. The sample data can be extended for further more days, so that more informative results can be introduced. Enhancement can be done by adding some more useful keywords. This work can be extended for health care, education system, etc. Other NLP techniques can also be used to increase the overall accuracy.

References

1. Alsudais, A., Leroy, G., Corso, A.: We know where you are tweeting from: assigning a type of place to tweets using natural language processing and random forests. In: 2014 IEEE International Congress on Big Data, Anchorage, AK, pp. 594–600, 27 June 2014–2 July 2014
2. Fu, C., Shi, G., Zhan, S.: A study on trend prediction in SinaWeibo Community. In: 2014 IEEE International Congress on Big Data, Anchorage, AK, pp. 364–365, 27 June 2014–2 July 2014
3. Achrekar, H., Gandhe, A., Lazarus, R., Yu, S.-H., Liu, B.: Predicting flu trends using twitter data. In: 2011 IEEE conference on computer communications workshops (INFOCOM WKSHPS), Shanghai, pp. 702–707, 10–15 Apr 2011
4. Alsultanny, Y.A.: Labour market forecasting by using data mining. In: 2013 International Conference on Computational Science, vol. 18, pp. 1700–1709 (2013)
5. Kannan, K.S., Sekar, P.S., Sathik, M.M., Arumugam, P.: Financial stock market forecast using data mining techniques. In: Proceedings of the International Multiconference of Engineers and Computer Scientists, pp. 555–559 (2010)
6. Lee, K., Agrawal, A., Choudhary, A.: Real-time disease surveillance using twitter data: demonstration on flu and cancer. In: Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (Oct 2011)
7. Prier, K.W., Smith, M.S., Giraud-Carrier, C., Hanson, C.L.: Identifying health-related topics on twitter an exploration of tobacco-related tweets as a test topic. In: 4th International Conference, SBP 2011, College Park, MD, USA, 29–31 Mar, 2011, pp. 18–25 (2011)
8. Hirose, H., Wang, L.: Prediction of infectious disease spread using twitter: a case of influenza. In: 2012 Fifth International Symposium on Parallel Architectures, Algorithms and Programming (PAAP), 17–20 Dec 2012, Taipei, pp. 100–105
9. Wang, H., Li, Y., Feng, Z., Feng, L.: ReTweeting analysis and prediction in microblogs: an epidemic inspired approach. Commun. China **10**(3), pp 13–24 (Mar 2013)
10. Java, A., Song, X., Finin, T., Tseng, B.: Why we twitter: understanding microblogging usage and communities. In: Proceedings of the Joint 9th WEBKDD and 1st SNA-KDD Workshop 2007, 12 Aug, 2007, pp. 56–65
11. NLTK Reference: http://www.nltk.org/api/nltk.html
12. Snowball Stemmer: http://snowball.tartarus.org/

Security Analysis of Scalable Speech Coders

Richa Tyagi, Kamini Malhotra and Anu Khosla

Abstract Scalable speech coders have been developed in order to achieve efficient speech communication over mobile ad hoc networks (MANETS). In practice, selective encryption is used with scalable speech coders to provide security of speech data. This paper analyzes the scalable ADPCM coders with selective encryption scheme where only core bits of coded speech are encrypted. The parameters of the coder have been identified by analyzing the selectively encrypted speech. The methodology developed is based on signal processing techniques. Autocorrelation method and power spectral density have been used to extract the bit rate and align the core and enhancement bits. Further zero crossing rate and Fourier Transform operations are used to detect speech-pause regions.

Keywords Scalable coder · Selective encryption · MANET · Core bits · Enhancement bits

1 Introduction

Scalable speech coders have been proposed in literature for voice communication over mobile ad hoc networks (MANETs). MANETs are self-configurable infrastructures less wireless networks providing mobility and ease of deployment. Achieving reliable and efficient voice communication over such wireless links is an important issue as packet loss rate and end-to-end packet delay tends to increase under adverse conditions of heavy traffic load, high mobility, and channel

R. Tyagi (✉) · K. Malhotra · A. Khosla
SAG, DRDO, Metcalfe House, Delhi, India
e-mail: richa.drdo@yahoo.co.in

K. Malhotra
e-mail: kaminimalhotra@sag.drdo.in

A. Khosla
e-mail: akhosla@sag.drdo.in

S. S. Agrawal et al. (eds.), *Speech and Language Processing for Human-Machine Communications*, Advances in Intelligent Systems and Computing 664,
https://doi.org/10.1007/978-981-10-6626-9_17

noise [1, 2]. One of the critical requirements for such networks is a bandwidth efficient speech coder which can perform well in spite of the constraints of MANETS.

Providing security to information communicated over such networks is an open research problem. One of the methods proposed for secure voice communication over these networks involves the combination of scalable speech coding with selective encryption of the bit stream [3–5]. This paper presents the security analysis of a combination of ADPCM (Embedded ADPCM—Telephone speech coder: G.727)-based scalable speech coder with selective encryption technique. Section 2 describes the scalable speech coder and selective encryption is discussed in Sect. 3. The analysis approach has been described in Sect. 4 and the Results and Conclusion have been discussed in Sect. 5.

2 Scalable Speech Coding

Mainly two types of scalable speech coders are available: SNR (Bit rate) and Bandwidth scalable coders [6, 7]. The SNR scalable coder provides acceptable quality of speech at lowest bit stream rate known as core bit stream rate. At the same time, it can also provide one or more enhancement bit streams which when combined with the core bit stream improves the speech quality at the receiver side. In bandwidth scalable coder, voice with a narrower bandwidth is coded as a base layer stream that provides acceptable speech quality. One or more enhancement layers that encode frequencies above the base layer bandwidth may be added to provide the improved speech quality.

There are two existing standards for SNR scalable coders, namely, G.727 which is based on embedded ADPCM coder and operates at rates of 16, 24, 32 and 40 kbps [8] and the MPEG-4 speech coding tool [9] based on code-excited linear prediction (CELP) with variable bit rates varying from 3.85 to 12.2 kbps.

In this paper, G.727 SNR scalable speech coder (Fig. 1) was considered for analysis. It is based upon embedded adaptive differential pulse code modulation and it operates at data rates of 16, 24, 32, and 40 kbps which correspond to 2, 3, 4, and 5 bits per sample, respectively. The rate of core layer is 16 kbps, and maximum of three enhancement layers each increasing the rate by 8 kbps can be included. The enhancement bit streams are dropped when wireless channel becomes congested, or to conserve mobile battery power [8].

The embedded ADPCM (G.727) [8] is an extension of basic ADPCM G.726 and is recommended for use in packetized speech systems operating according to the Packetized Voice Protocols (PVP). The PVP is able to relieve congestion by modifying the size of packet when the need arises. Table 1 defines the combination of core and enhancement bits for different bit rates [8].

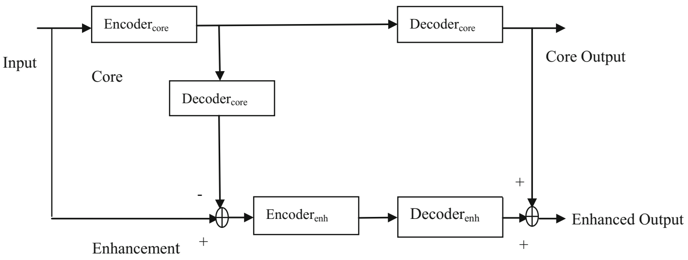


Fig. 1 Basic diagram of scalable speech coder

Table 1 Combination of core and enhancement bits for different bit rates [8]

Coding rate (kbps)	Core bits (C)	Enhancement bits (E)
16	2	0
24	2 3	1 0
32	2 3 4	2 1 0
40	2 3 4	3 2 1

3 Selective Encryption

Selective encryption is a technique where only some of the bits of the data are selected and encrypted while rest of the bits are transmitted in clear in order to conserve resources such as bandwidth and battery power [10] (Fig. 2).

The important issue is to decide which bits are to be protected to provide expected security so that if any security attack happens on the unprotected bits the adversary should not be able to detect any important information about transmitted data. While several studies of selective encryption, based on different types of block and stream ciphers, for video and image compression have been performed and documented [11, 12], very few results on selective encryption of coded speech have been presented [3, 4].

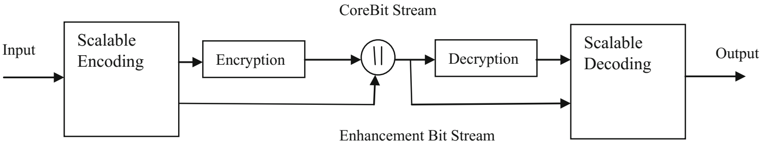


Fig. 2 Scalable coding with selective encryption

Servetti and De Martin [3] investigated partial encryption of non-scalable coder G.729 at 8 kbps and demonstrated that partial encryption of about 45% of the bit stream provides protection equivalent to full encryption. Gibson and Servetti [4] investigated the performance of selective encryption on MPEG-4 standard CELP scalable speech coding tool. It is claimed that as the enhancement bits only provide quality and as such do not leak any information about the message, encryption of the core layer of scalable coders only is sufficient to ensure a high level of protection.

In this paper, selective encryption of G.727 using stream cipher [13, 14] has been analyzed and it has been shown that the partially encrypted coded speech leaks information about the parameters as well as underlying speech.

4 Analysis of Selectively Encrypted Scalable Speech Coding

The objective of the analysis was to determine the strength of selective encryption over G.727 coding. For the analysis, G.727 scalable coder with selective encryption was implemented in software and analysis was carried out using signal processing techniques. The input PCM speech data at 64 kbps was encoded at variable rates (40, 32, 24 and 16 kbps) and the core bits of the encoded bit stream were then selectively encrypted by XORing it with linear feedback shift register sequence. The encrypted bit stream was then intercepted like an eavesdropper. The spectrogram of the input, encoded and encrypted (core bits—encrypted and enhancement bits—clear) speech is shown in Fig. 3.

The complete methodology of analysis various operations carried out on the encrypted speech signal is described in the flowchart given in Fig. 4.

4.1 Identification of Bit Rate

The coder has four variable bit rates as mentioned so the first step is to find out the bit rate/bits per sample in the intercepted data. The autocorrelation method which measures the similarity between intercepted sequence and its time-delayed version was used.

By taking sample to sample autocorrelation, a peak after the actual number of bits per sample was observed because it attains a maximum at samples 0, ±P, ±2P…, if signal is periodic with period *k* samples. The plot of the autocorrelation function output is shown in Fig. 5.

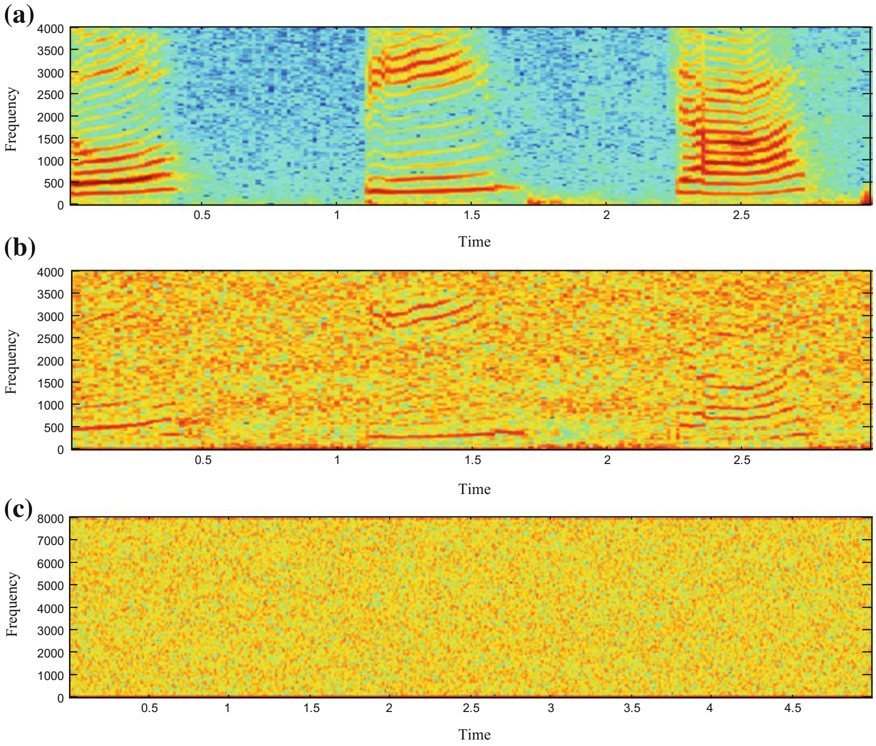


Fig. 3 Spectrogram of **a** input **b** encoded and **c** encrypted speech

4.2 *Identification and Alignment of Core and Enhancement Bits*

Next step was to identify core and enhancement bits (C, E) and correct alignment of the given bit stream. The power spectral density (PSD) function which estimates the distribution (over frequency) of the power contained in the given signal was used for this purpose. The following steps were executed to obtain the accurate result:

1. The power spectral density of the given bit stream was computed in a sliding fashion for all combination of core and enhancement bits, e.g., let the intercepted signal have bit rate of 40 kbps with 5 bits/sample and core and enhancement combination is (2, 3).
2. The difference of core and enhancement bits PSD range was calculated for all combinations.
3. Maximum value of difference was computed as per Eq. 1:

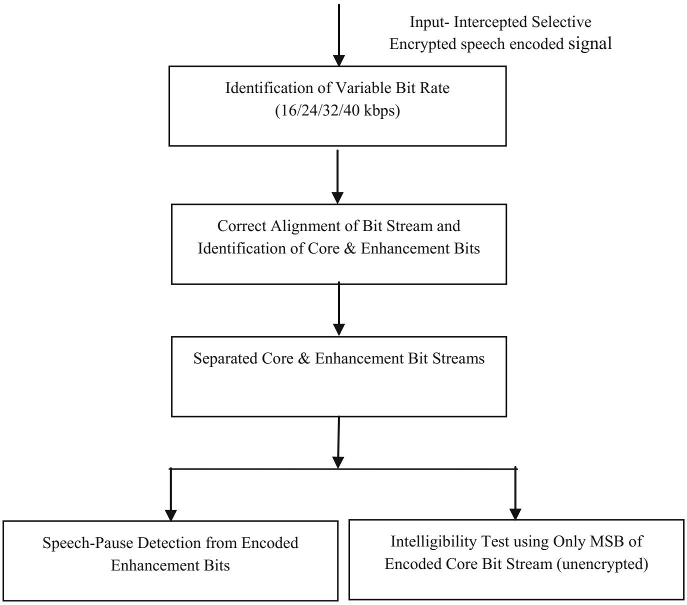

Input- Intercepted Selective
Encrypted speech encoded signal
Identification of Variable Bit Rate
(16/24/32/40 kbps)
Correct Alignment of Bit Stream and
Identification of Core & Enhancement Bits
Separated Core & Enhancement Bit Streams
Speech-Pause Detection from Encoded
Enhancement Bits
Intelligibility Test using Only MSB of
Encoded Core Bit Stream (unencrypted)

Fig. 4 Analysis flowchart of the complete methodology

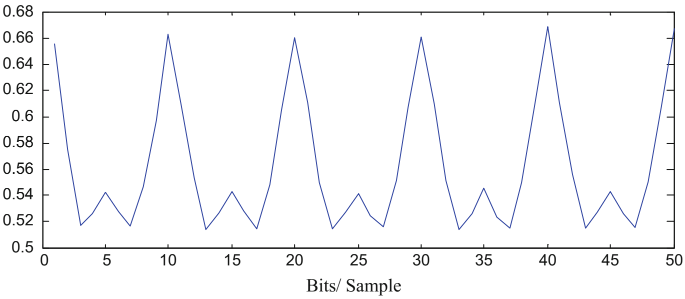


Fig. 5 Autocorrelation's plot of selectively encrypted ADPCM coded speech, 5-bit scalable coder

$$D_{\max} = \mathrm{Max}|(\mathrm{Range}(\mathrm{PSD}(c_i)) - \mathrm{Range}(\mathrm{PSD}(e_i)))|, \quad i = 1\,\mathrm{to}\,5 \tag{1}$$

PSD (c_i) and PSD (e_i) being the power spectral density value for core and enhancement bits for each combination.

4. Steps 1–3 were repeated for all combinations of core and enhancement, i.e., (2, 3), (3, 2), (4, 1) (three combinations), and $D_{\max}$ computed for all combinations.
5. Maximum of the $D_{\max}$ in the above-computed values indicated the correct alignment of bit stream.

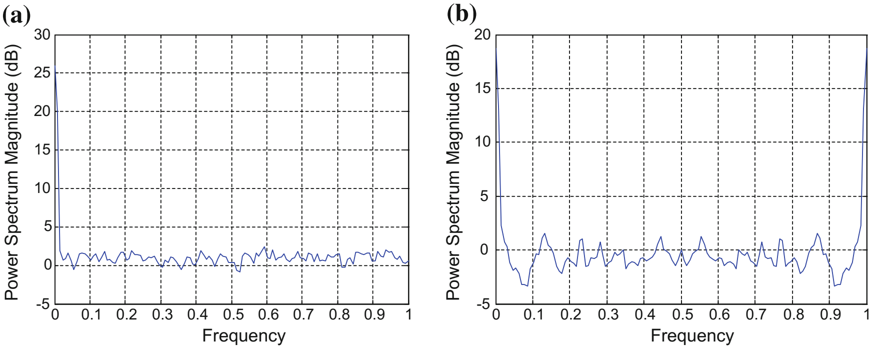


Fig. 6 PSD of **a** encrypted core bits and **b** encoded enhancement bits in correctly aligned frame

This maximum difference is obtained for the correctly aligned bits as the power spectral density of the encrypted core bits is flat (similar to a random noise) whereas for other combinations power spectral density of core bits is having some periodic variations. The difference can be easily seen as depicted in Fig. 6.

4.3 Speech-Pause Detection from Encoded Enhancement Bits

The input data was segmented into 32 ms frames, and zero crossing rate and Fast Fourier Transform (FFT) were computed from the unencrypted enhancement bits.

The spectrogram of input speech and plain enhancement bits is shown in Fig. 7. The zero crossing rate obtains lower values in speech regions as compared to pause regions as shown in Fig. 8. As the magnitude of pause region is high as compared to speech region in the enhancement layer so it gives higher value of FFT coefficients for pause regions as shown in Fig. 8. The speech-pause regions were separated based on a threshold value.

4.4 Intelligibility Tests

In G.727 scalable coder, the minimum bit rate is specified as 16 kbps and minimum two core bits are required for transmission that provides acceptable coded speech quality. Only encoded core bit (unencrypted) frequency pattern was studied, and after analyses, it was concluded by listening tests that only first most significant bit (MSB) of encoded speech can also lead to intelligible speech. Some formants are visible in the spectrogram of single core bit (MSB) speech is shown in Fig. 9.

Fig. 7 Spectrogram of **a** input speech and **b** encoded enhancement layer

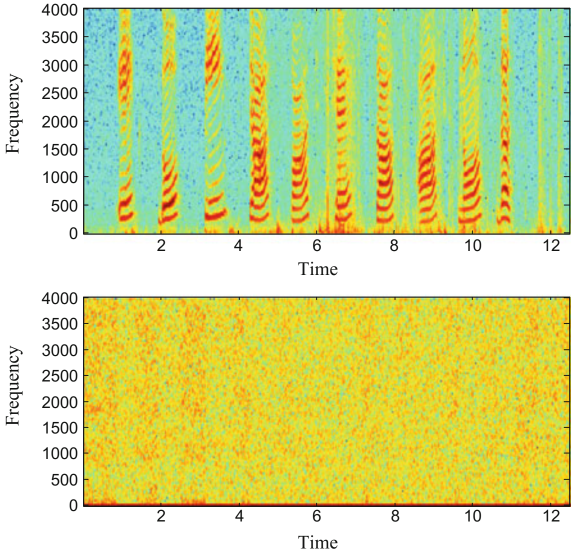
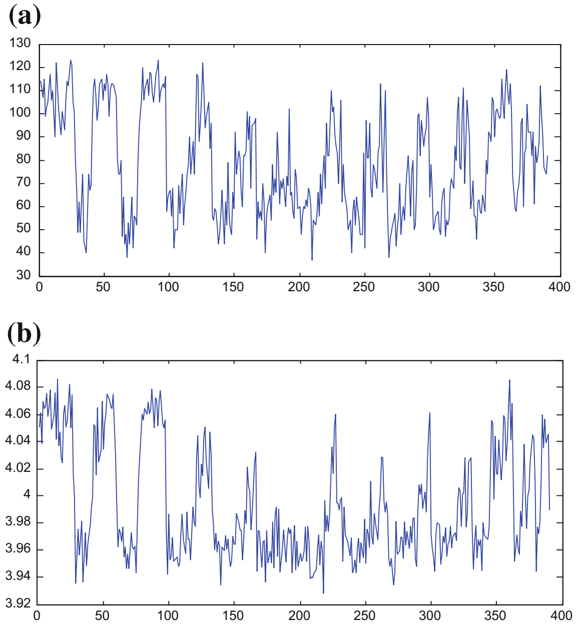


Fig. 8 Plot of **a** Zero crossing rate and **b** FFT



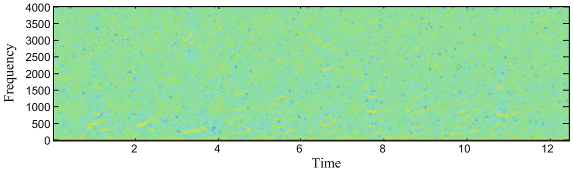


Fig. 9 Spectrogram of single core bit (MSB) speech

5 Conclusion and Discussions

An approach to analyze security of selectively encrypted speech encoded using scalable ADPCM coding transmitted over ad hoc networks has been proposed. It is shown that encryption of only core bit stream of scalable embedded ADPCM speech coder (G.727) as proposed in literature is not sufficient to ensure security. Simple signal processing techniques like autocorrelation, power spectral density, zero crossing rate, and Fourier Transform can be applied to extract useful information from such encrypted signals. The knowledge of pause regions obtained from analysis of enhancement bits may lead to recovery of partial key bits. The intelligibility test performed on unencrypted core bit stream indicated that recovery of only the MSB can result in intelligible speech.

References

1. Siva Ram Murthy, C., Manoj, B.S..: Ad Hoc Wireless Networks Architectures and Protocols. Pearson Education (2004)
2. Earle, A.E.: Wireless Security Handbook. Auerbach Publication (2006)
3. Servetti, A., De Martin, J.C.: Perception-based partial encryption of compressed speech. IEEE Trans. Speech Audio Process. **10**, 637–643 (2002)
4. Gibson, J.D., Servetti, A., Dong, H., Gersho, A., Lookabaugh, T., De Martin, J.C.: Selective encryption and scalable speech coding for voice communication over multi-hop wireless links. In: IEEE Military Communication Conference (MILCOM), pp. 792–798 (2004)
5. Dong, H., Chakares, I.D., Lin, C.H., Gersho, A., Belding-Royer, E., Madhow, U., Gibson, J. D.: Speech Coding for Mobile Ad hoc Networks. Research work supported by NSF, University of California
6. Dong, H., Gibson, J.D.: Structure for SNR scalable speech coding. IEEE Trans. ASL Process. **14**(2) (2006)
7. Dong, H.: SNR and Bandwidth Scalable Speech Coding. Ph.D. thesis, Southern Methodist University, Dalas, Texas (2002)
8. ITU-T: 5-, 4-, 3- and 2-bit/ sample Embedded Adaptive Differential Pulse Code Modulation (ADPCM) (1990)
9. ISO/IEC JTCI SC29/WG11, ISO/IEC FDIS 14496-3: Information Technology-Coding of Audiovisual Objects- Part 3: Audio (1998)
10. Meshram, P., Bhaisare, P., Karale, S.J.: Comparative study of selective encryption algorithm for wireless adhoc network. In: JREAS, vol. 2, pp. 506–517 (2012)
11. Cheng, H., Li, X.: Partial encryption of compressed images and videos. IEEE Trans. Sig. Process. **3**, 2439–2451 (2000)

12. Alattar, A.M., AI-Regib, G.I.: Evaluation of selective encryption techniques for secure transmission of MPEG-compressed bit-stream. In: IEEE International Symposium on Circuits and Systems, pp. 340–343 (1999)
13. Menezes, A., Van Oorschot, P., Vanstone, S.: Handbook of Applied Cryptography. CRC Press, Boca Raton (1996)
14. Stallings, W.: Cryptography and Network Security: Principles and Practice, 5th edn. Pearson Education (2011)

Issues in i-Vector Modeling: An Analysis of Total Variability Space and UBM Size

Mohit Kumar, Dipangshu Dutta and Pradip K. Das

Abstract Recent trends have indicated the use of very high computations for solving the problem of speaker recognition. However, there are cases when gains are not commensurate to the additional computations involved. We have studied the effect of size of UBM and the total variability matrix, T, in i-vector modeling on the recognition performance. Results indicate that after T size 50, there is a very small performance improvement. For UBM size, 128 is observed as the optimal mixture count. For performing the experiments, we have used the ALIZE toolkit and TED-LIUM database.

Keywords i-vectors · Joint factor analysis · GMM · UBM · JFA

1 Introduction

Research on speaker identification and verification has been going on since decades. However, a definite solution to the problem is still not found. Speaker identification is the process of identifying the person who produced the utterance under test. Speaker verification is the task of checking whether the test utterance is produced by the claimed speaker or not. The decision is either a yes or a no.

There have been several attempts to solve the problem of speaker identification. This work has been first reported in [1] with the introduction of Gaussian mixture models (GMMs) to model the speaker using the mixture components. The means and the variances thus obtained were used as models. The identification was done by finding the probability of the utterance given in the models, and the model with

M. Kumar (✉) · D. Dutta · P. K. Das
Department of CSE, IIT Guwahati, Assam 781039, India
e-mail: mohit.2013@iitg.ernet.in

D. Dutta
e-mail: dipangshu@iitg.ernet.in

P. K. Das
e-mail: pkdas@iitg.ernet.in

S. S. Agrawal et al. (eds.), *Speech and Language Processing for Human-Machine Communications*, Advances in Intelligent Systems and Computing 664,
https://doi.org/10.1007/978-981-10-6626-9_18

highest probability was selected as the identified speaker. In order to improve convergence time and generate better models, adapted GMMs were proposed [2]. The principal idea was same as the original GMM model but with the modification that the speaker models were now not generated from scratch. Instead, universal background model (UBM) was used for generating speaker models.

In 2005, the concept of joint factor analysis was introduced [3, 4]. The idea was to model the variability of speaker and channel effects separately. However, i-vector modeling was soon proposed as an improvement to JFA modeling.

The i-vector paradigm has been an active area in speaker recognition and verification research [5]. Dehak et al. [6] proposed a channel-blind approach for telephone as well as microphone data. References [7, 8] have presented ways to calculate i-vectors in an efficient way. The use of PLDA for channel and speaker compensation has been explained in [9]. At the same time, [10] demonstrated that if training utterances are sufficiently long, shorter utterances for testing can also give better accuracy. Reference [11] showed that PLDA is useful even when the testing utterances length is variable. Reference [12] has tried to reduce the data requirement for training by using k-nearest neighbor (k-NN) algorithm. Reference [13] has compared robustness of different approaches to speaker recognition. I-vectors have also been employed in language recognition tasks [14, 15]. Reference [16] has used extended feature vector, consisting of MFCCs with some tandem features, before calculation of i-vectors. They have discovered the differences in features to be used for speaker and language recognition tasks.

The rest of the paper is organized as follows. The next section describes the motivation behind the study. The third section highlights upon the speaker modeling techniques used and the theory behind speaker recognition systems. The fourth section describes the experimental setup and data used. The fifth section reports the results obtained from the experiments and discusses important observations. We conclude the paper in the sixth section.

2 Motivation

Several studies have been conducted suggesting the use of i-vector-based methods to solve the problem of speaker identification. Many studies also experimented with different compensation techniques to be used along with the method. However, it is not explicitly formulated why a particular configuration of the method should be used. The motivation behind the following experiments is to determine the optimal size of the total variability matrix, T, which should be used for generating i-vectors so that the computation involved can be reduced to minimum, along with maintaining acceptable accuracy. It is also worth exploring how, for a particular size of T, the number of mixture in the UBM affects the accuracy. Another interesting variable is the size of the test utterance. Exploring this issue is also one of the motivations behind this study.

3 Speaker Modeling Techniques

GMMs have been used for speaker recognition tasks for more than two decades now [1]. They have served as the basis for speaker recognition experiments. They have the ability to form smooth approximations to arbitrary densities. A GMM is a weighted sum of C components. It is represented as follows:

$$p(x|\lambda) = \sum_{i=1}^{C} w_i g\left(x|\mu_i, \sum\nolimits_i\right)$$

where x is a D-dimensional vector, w_i, $i = 1, 2, \ldots, C$ are the mixture weights, and $g\left(x|\mu_i, \sum_i\right)$, $i = 1, 2, \ldots, C$ are the component densities of the Gaussian. The intuition is that the components of the Gaussian may in some way represent speaker's broad phonetic events [2].

To handle data constraints and enable faster convergence, adapter GMMs were proposed. Instead of building speaker models from scratch, UBM parameters were used to generate speaker models, and MAP algorithm was used for adaptation. The adaptation parameters were used to control the balance of UBM characteristics and speaker utterance characteristics.

Kenny et al. [3] proposed a joint factor analysis model to explain most of the variance of speaker and channel through smaller number of factors. The model aims to make a decision that whether the difference in the test and reference utterances can be accounted by inter-speaker variability or intra-speaker variability. Intra-speaker variability may arise due to channel and speaker's emotional state. It was widely understood that speaker variability modeling is more important than modeling channel variability. However, in an analogous study [17] in face recognition, it was found that modeling intra-person variability, without modeling inter-person variability, led to good performance. The factor analysis can be represented as follows:

$$M = m + V_y + U_x + D_z$$

where m is a speaker and session independent supervector of means which is obtained generally from the UBM means, $V \& D$ are related to speaker subspace, and U reflects the session space.

Dehak et al. [18] showed that even while modeling channel factors and speaker factors separately, there was still some information about speaker in the channel factors. A single total variability space was therefore proposed. The total variability space is then used to extract i-vectors. In i-vector modeling, a GMM supervector is represented as follows:

$$M = m + Tw$$

where m is a speaker and channel independent supervector, T is a low-rank matrix which represents the principal directions of the speaker and channel variability and is the basis of the smaller total variability space, w is a standard normal vector. The components of w are called total factors, and they represent the coordinates of the speaker in the reduced total variability space.

4 Experimental Setup/Layout

4.1 Data/Corpus

For our experiments, we used a part of the TED-LIUM [19] dataset/database. The TED-LIUM dataset consists of 1495 TED talk recorded at a sample rate of 16,000, 16 bit representation, with a bit rate of 256 K. The encoding used is 16-bit signed integer PCM. Out of the 1495 recordings, we used 50 of them in our experiments which amount to more than 10 h of speech data. For preprocessing, we removed the initial music found in all the data files. We split the data into two parts: training and testing. For training, we used approximately 15 min of the recording for each speaker. The testing part was different for each of the experiments conducted.

For the first experiments, we used utterances of 5 s in length as test cases. Several such tests were conducted for each speaker resulting in at least 40 test cases for each speaker. The total test cases used were 2819.

For the second and third experiments, we used utterances of length 10 and 15 s, respectively. The fourth test used the complete test files which were of length at least 3 min and at most 8 min. These tests were designed to understand how the length of the test utterance affects the accuracy of the recognition system.

4.2 Feature Extraction

Feature extraction is an important step in any recognition system. For our experiments, we used the SPro tool to extract the features. We used the 60-dimensional feature vector for each frame. Here, each frame refers to a segment of 10 ms from the utterance. The feature set used is mel-frequency cepstrum coefficients [20, 21]. The 60-dimensional feature vectors consist of 19 static coefficients followed by log energy value, followed by the delta and acceleration coefficients.

4.3 Experiments

We experimented with different sizes of the total variability matrix for a particular mixture count in the UBM. We started with 16 mixtures in the UBM and increased the mixture count up to 512. We also tested that the performance of system by keeping the i-vector size constant while varying the mixture count in the UBM. Further, we tested the system with varying length of test utterances. For performing our experiments, we used the ALIZE toolkit [22]. For scoring, we have used the cosine distance scoring method.

5 Results and Discussion

In case of 5 s length test utterances, we found that when we vary the size of the total variability matrix while keeping the mixture count of the UBM fixed, we see a distinct rise in accuracy when mixture count is increased from 16 to 32. However, there is a significant dip at 64 mixtures except when the T matrix size is 200. This indicates that for representing the speaker characteristics, 64 may not be a good mixture count for most of the cases. On increasing the number of mixtures to 128, there is further increase in accuracy but the improvement from 32 to 128 mixtures is less than the improvement observed in going from 16 to 32 mixtures. Increasing the mixture count to 256 and 512 components does not yield significant improvements. This indicates that after 128 mixtures, we are using a large amount of computation for very small improvements. Thus, it would be useful to choose 128 mixtures for speaker representation. However, 32 mixture count can also serve well in cases where there are computational limitations (Figs. 1, 2, 3, 4, 5, 6, 7, 8, and 9; Table 1)

We also observed that if we increase the size of the T matrix while keeping the number of mixtures in the UBM constant, there are no significant gains after the size 50. There is, however, a jump in accuracy of one case when the mixture count is 64. In that case, we see accuracy going from 79.60% at size 50–90.28% at size 200. Therefore, T size 50 may be proposed as an acceptable working configuration. Along with this, the mixture count may be chosen as 32 or 128, depending on the availability of computation resources.

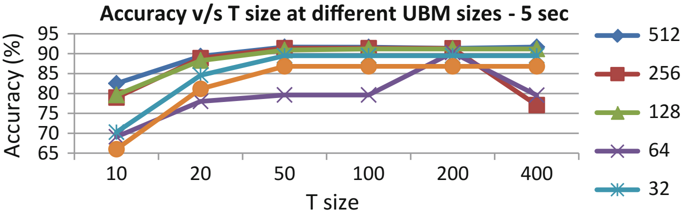


Fig. 1 Performance with 5 s test utterance at different UBM sizes

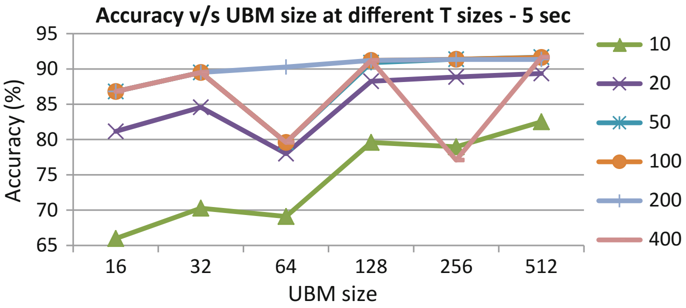


Fig. 2 Performance with 5 s test utterance at different T sizes

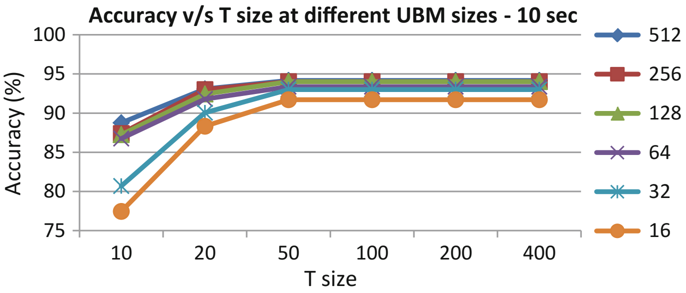


Fig. 3 Performance with 10 s test utterance at different UBM sizes

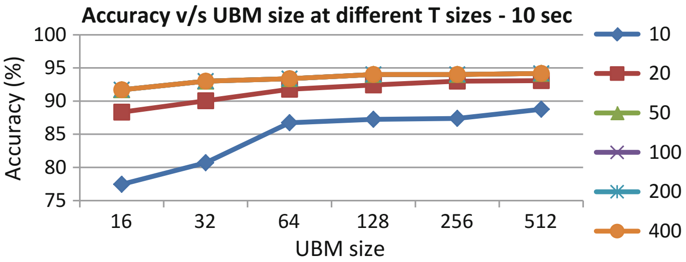


Fig. 4 Performance with 10 s test utterance at different T sizes

Similar trends were seen for 10, 15 s, and full-length test files. Another interesting observation from the results is that when we increase the length of test utterances, performance increases from 5 s length to 10 s length. However, there is either a dip or no improvement in performance while going from 10 to 15 s utterances. We also see that the performance is 100%, except for the case of T size 10, for complete length test files. This is expected as the test utterances have sufficient amount of data in them for testing accurately.

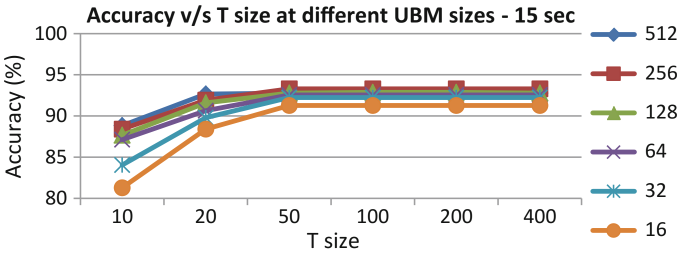


Fig. 5 Performance with 15 s test utterance at different UBM sizes

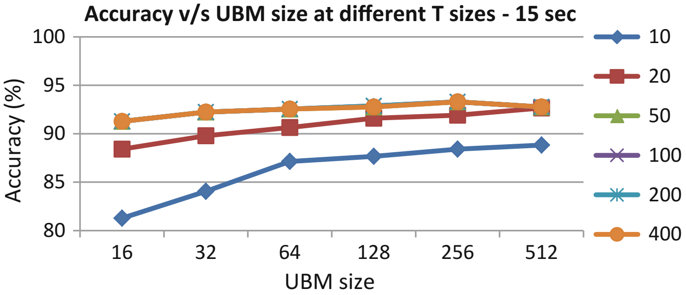


Fig. 6 Performance with 15 s test utterance at different T sizes

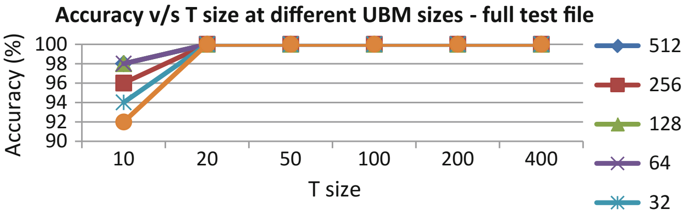


Fig. 7 Performance with full-length test utterance at different UBM sizes

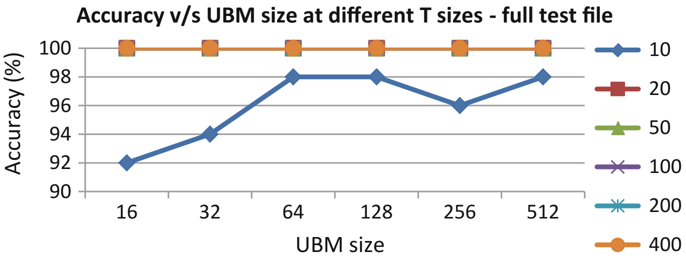


Fig. 8 Performance with full-length test utterance at different T sizes

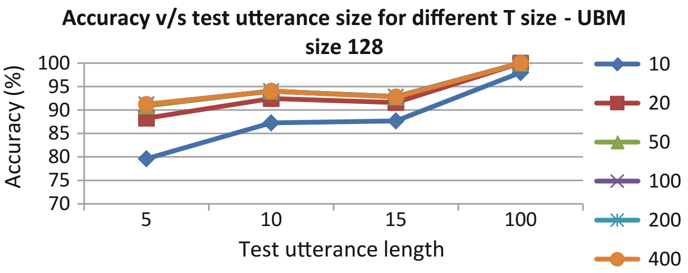


Fig. 9 Performance with different length tests utterance for UBM size 128

Table 1 Performance evaluation with 5, 10, 15 s, and full test files

No of mixtures	T size	Accuracy (5-s)	Accuracy (10-s)	Accuracy (15-s)	Accuracy (full test file)
512	10	82.511529	88.76081	88.82979	98
512	20	89.357928	93.08357	92.65957	100
512	50	91.663711	94.16427	92.76596	100
512	100	91.628237	94.16427	92.76596	100
512	200	91.344448	94.16427	92.76596	100
512	400	91.663711	94.16427	92.76596	100
256	10	78.964172	87.39193	88.40426	96
256	20	88.861298	93.01153	91.91489	100
256	50	91.379922	94.02017	93.29787	100
256	100	91.379922	94.02017	93.29787	100
256	200	91.344448	94.02017	93.29787	100
256	400	77.084072	94.02017	93.29787	100
128	10	79.602696	87.24784	87.65957	98
128	20	88.258248	92.43516	91.59574	100
128	50	90.883292	94.02017	92.76596	100
128	100	91.202554	94.02017	92.87234	100
128	200	91.202554	94.02017	92.87234	100
128	400	91.202554	94.02017	92.76596	100
64	10	69.102519	86.74352	87.12766	98
64	20	78.006385	91.78674	90.6383	100
64	50	79.602696	93.37176	92.55319	100
64	100	79.602696	93.37176	92.55319	100
64	200	90.280241	93.37176	92.55319	100
64	400	79.567222	93.37176	92.55319	100
32	10	70.273147	80.69164	84.04255	94
32	20	84.568996	90.05764	89.78723	100
32	50	89.499823	93.01153	92.23404	100

(continued)

Table 1 (continued)

No of mixtures	T size	Accuracy (5-s)	Accuracy (10-s)	Accuracy (15-s)	Accuracy (full test file)
32	100	89.499823	93.01153	92.23404	100
32	200	89.499823	93.01153	92.23404	100
32	400	89.499823	93.01153	92.23404	100
16	10	65.980844	77.44957	81.2766	92
16	20	81.163533	88.32853	88.40426	100
16	50	86.803831	91.7147	91.2766	100
16	100	86.803831	91.7147	91.2766	100
16	200	86.803831	91.7147	91.2766	100
16	400	86.803831	91.7147	91.2766	100

6 Conclusion and Future Work

In this work, we analyzed the effect of variation of size of the total variability matrix, T, for a given mixture count, on the accuracy of the i-vector system. We also analyzed how the mixture count of UBM affects the accuracy while keeping the size of T matrix constant. We found that the T matrix size 50 along with mixture count of 32 or 128 shows good performance. This also helps to reduce computational costs. While studying the effect of length of test utterances on performance, we observed that the 10 s length utterances produced the best results. However, 100% accuracy can be achieved with longer test utterances. Interesting future experiments can include the analysis of the length of size of training utterances on the performance of the system.

References

1. Reynolds, D.A., Rose, C.R.: Robust text-independent speaker identification using Gaussian mixture speaker models. IEEE Trans. Speech Audio Process. **3**(1), 72–83 (1995)
2. Reynolds, D.A., Quatieri, T.F., Dunn, R.B.: Speaker verification using adapted Gaussian mixture models. Digit. Sig. Process. **10**(1), 19–41 (2000)
3. Kenny, P., Boulianne, G., Dumouchel, P.: Eigenvoice modeling with sparse training data. IEEE Trans. Speech Audio Process. **13**(3), 345–354 (2005)
4. Kenny, P., Boulianne, G., Ouellet, P., Dumouchel, P.: Jointfactor analysis versus eigenchannelsin speaker recognition. IEEE Trans. Audio Speech Lang. Process. **15**(4), 1435–1447 (2007)
5. Verma, P., Das, P.K.: i-Vectors in speech processing applications: a survey. Int. J. Speech Technol. 18:1381–2416 (2015)
6. Dehak, N., Karam, Z.N., Reynolds, D.A., Dehak, R., Campbell, W.M., Glass, J.R.: A channel-blind system for speaker verification. In: 2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 4536–4539 (2011)

7. Glembek, O., Burget, L., Matejka, P., Karafiat, M., Kenny, P.: Simplification and optimization of i-vector extraction. In: 2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 4516–4519 (2011)
8. Aronowitz, H., Barkan, O.: Efficient approximated i-vector extraction. In: 2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 4789–4792 (2012)
9. Jiang, Y., Lee, K.A., Tang, Z., Ma, B., Larcher, A., Li, H.: PLDA modeling in i-vector and supervector space for speaker verification. In: INTERSPEECH-2012, pp. 1680–1683 (2012)
10. Sarkar, A.K., Matrouf, D., Bousquet, P.M., Bonastre, J.F.: Study of the effect of i-vector modeling on short and mismatch utterance duration for speaker verification. In: INTERSPEECH-2012, 2662–2665 (2012)
11. Kenny, P., Stafylakis, T., Ouellet, P., Alam, M.J., Dumouchel, P.: PLDA for speaker verification with utterances of arbitrary duration. In: 2013 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 7649–7653 (2013)
12. Biswas, S., Johan R., Koichi S.: i-Vector selection for effective PLDA modeling in speaker recognition. In: Proceedings of Odyssey Workshop, ISCA, pp. 100–105 (2014)
13. Mandasari, M.I., McLaren, M., van Leeuwen, D.A.: The effect of noise on modern automatic speaker recognition systems. In: 2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 4249–4252 (2012)
14. Martínez, D., Plchot, O., Burget, L., Glembek, O., Matějka, P.: Language recognition in ivectors space. In: INTERSPEECH-2011, pp. 861–864 (2011)
15. Dehak, N., Torres-Carrasquillo, P.A., Reynolds, D.A., Dehak, R.: Language recognition via i-vectors and dimensionality reduction. In: INTERSPEECH-2011, pp. 857–860 (2011)
16. Li, M., Liu, W.: Speaker verification and spoken language identification using a generalized i-vector framework with phonetic tokenizations and tandem features. In: INTERSPEECH-2014, pp. 1120–1124 (2014)
17. Slomka, S., Castellano, P., Barger, P., Sridharan, S., Narasimhan, V.L.: A comparison of Gaussian mixture and multiple binary classifier models for speaker verification. In: Australian and New Zealand Conference on Intelligent Information Systems, 1996, pp. 316–319 (1996)
18. Dehak, N., Kenny, P., Dehak, R., Dumouchel, P., Ouellet, P.: Front-end factor analysis for speaker verification. IEEE Trans. Audio Speech Lang. Process. **19**(4), 788–798 (2011)
19. Rousseau, A., Deléglise, P.,Estève, Y.: Enhancing the TED-LIUM corpus with selected data for language modeling and more TED talks. In: Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14) (2014)
20. Liu, Q., Sung, A., Qiao, M.: Temporal derivative-based spectrum and mel-cepstrum audio steganalysis. IEEE Trans. Inf. Forensics Secur. **4**(3), 359–368 (2009)
21. Sharma, S., Kumar, M., Das, P.K.: A technique for dimension reduction of MFCC spectral features for speech recognition. In: International Conference on Industrial Instrumentation and Control 2015, pp. 99–104 (2015)
22. Larcher, A., Bonastre, J., Fauve, B.G.B., Lee, K., Levy, C., Li, H., et al.: ALIZE 3.0: open source toolkit for state-of-the-art speaker recognition. In: INTERSPEECH 2013, pp. 2768–2772 (2013)

Acoustic Representation of Monophthongs with Special Reference to Bodo Language

Uzzal Sharma

Abstract Speech recognition is getting popularity day by day, as it plays a remarkable role in the field of human–computer communication (HCC). Due to the inclusion of speech recognition feature in the latest operating systems, its importance has been increased many folds. Although a lot of work has been conducted in English, the other languages have not yet been completely explored as far as speech research is concerned. In the present paper, a major language of northeast India, the Bodo language, has been studied in terms of monophthong sounds present in it, using formant frequency based on LPC. The study reveals a number of facts which will be helpful in the speech and speaker recognition.

Keywords HCC · LPC · Formant · Monophthong · Diphthongs · Vowel · Consonants

1 Introduction

The pure vowels of any language are known as monophthong. The articulation of monophthong does not glide up or down toward a new articulation position during the articulation. Their articulation is almost fixed at the beginning and end of articulation. In contrast, spoken language also has another type of sound pattern known as diphthong. The diphthongs are characterized by the presence of two vowel sounds in it. Practically, all the diphthongs contain two monophthongs. For example in English, the word "bit" has a monophthong having only one vowel sound; on the other hand, the word "tear" has a diphthong having two vowel sounds, where it glides from one vowel sound to another. Similarly, in case of Bodo language, the words Goi, aai, jiu, eo, khao are diphthong type words [1]. Some of the Bodo monophthong words are si, er, su, ga, ran. Although Bodo is a tonal language, in the current study, the tonal aspect is not considered [1]. In the current

U. Sharma (✉)
School of Technology, Assam Don Bosco University, Guwahati, India
e-mail: druzzalsharma@gmail.com

S. S. Agrawal et al. (eds.), *Speech and Language Processing for Human-Machine Communications*, Advances in Intelligent Systems and Computing 664,
https://doi.org/10.1007/978-981-10-6626-9_19

paper, the monophthongs of Bodo language will be studied from the acoustic view point, so that it will become an important cue toward the study of Bodo language for the purpose of speech recognition. The articulated sounds are categorized as voiced and unvoiced. The monophthongs or vowel sounds fall under voiced sound.

2 Background

It has already been established that a sort segment of speech can be considered as the output of a stationary or quasistationary wave. The speech signal can also be considered as the output of a linear system which is fixed for a time interval, excited as a quasiperiodic series of impulse train [2]. All the formant frequencies do not contain meaningful information which is sufficient enough to distinguish the voiced sounds or vowels. But the first four formant frequencies, namely F1, F2, F3, and F4 contain important information which is capable enough to distinguish the voiced sound or vowels [3]. But in the current study, we are going to consider only first three formants.

Bodo language is derived from Tibeto-Burman class which is in turn derived from Sino-Tibetan language group. Bodo language is one of the scheduled languages spoken in India especially in the northeastern part of India which has given special rights by the constitution of India. It is the official language of BTC, Assam. There are six pure vowels, sixteen consonants including two semivowels [4, 5–6] in Bodo language as shown in the following tables (Tables 1 and 2) respectively.

3 Recording Setup and Laboratory Environment

The environment and laboratory setup in which the experiment is conducted is very important as far as the accurate result is concerned. That is why, it is very much important to conduct the experiment in a perfect environment, which is globally standardized [7] [8]. In the current research, the following setup has been employed to maintain a standard accepted by majority:

- Microphone shure: model number SM63,
- Frequency response: 80 Hz–20 kHz,
- 5-channel audio mixer 20 Hz–20 kHz, model UNISOUN, UB-100e,

Table 1 Vowels present in Bodo language (pure)

	Tongue position		
	Front	Central	Back
Close	i		ɥ, ɯ
Mid	e		ɔ
Open		ə	

Table 2 Consonants present in Bodo language

Articulation type	Bilabial	Alveolar	Palatal	Velar	Glottal
Plosive	b	d		g	
	p^h	t^h		k^h	
Nasal	m	n		ɳ	
Fricative		s z			h
Trill		r			
Lateral		l			
Semivowel	w		y		

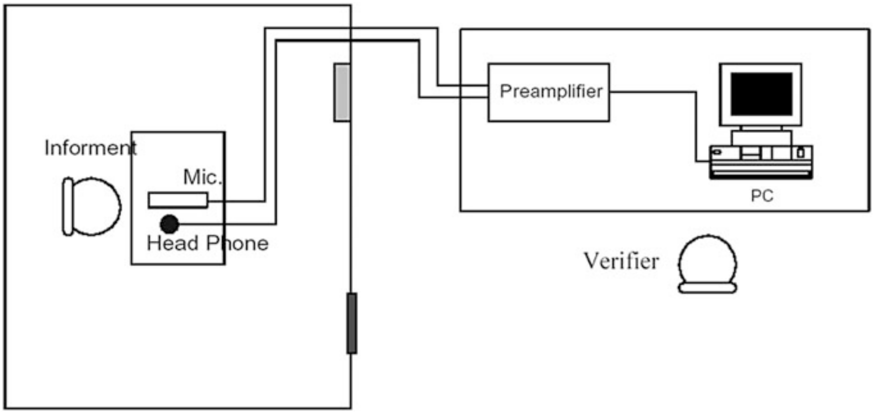


Fig. 1 Block diagram of recording process

- Sound card: good quality (creative make),
- Distance maintained 8 cm between the speaker and the microphone.

The setup diagram of the recording process is shown in Fig. 1.

4 Experiment

In the current study, the formants of six pure vowels of Bodo language will be calculated and analyzed for 50 male and 50 female, so that some important information can be achieved for further use [9–11]. For this, the formant frequencies of six pure vowels will be estimated using linear predictive coding (LPC), by using the value of linear prediction coefficient. The formant frequencies will be obtained from the calculation of the roots of the prediction polynomial [1]. The different steps involved are as follows which is diagrammatically represented in Fig. 2.

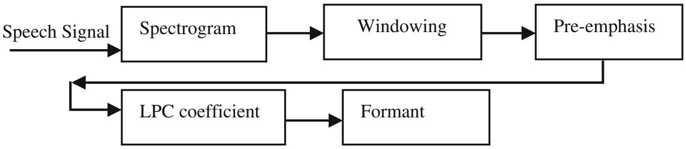
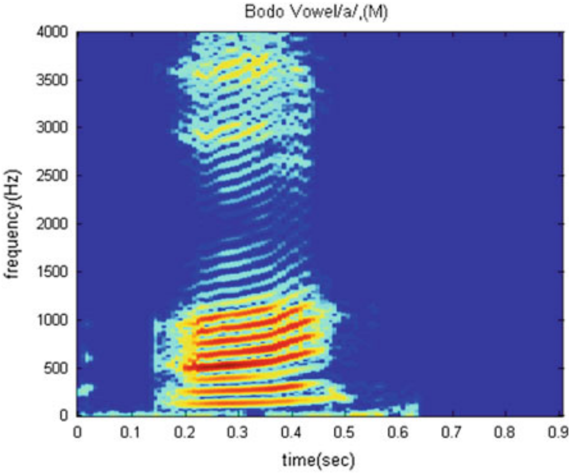


Fig. 2 Formant estimation

Fig. 3 Spectrogram for the vowel /a/



i. As a first step, the voiced segment of pure vowels is identified by using spectrogram function. The appropriate segment is extracted from the spectrogram for the purpose of analysis. After this, the speech segment is subjected to preprocessing Fig. 3.

ii. As a part of preprocessing of the segment, first the speech segment is subjected to windowing followed by another step known as pre-emphasis.

iii. After the windowing and pre-emphasis, the linear prediction coefficients (Eq. 1) can be obtained.

iv. Finally, the formants can be generated based on the linear prediction coefficient.

$$\hat{\boldsymbol{x}}(\boldsymbol{n}) = \sum_{i=1}^{p} \boldsymbol{a}_i \boldsymbol{x}(\boldsymbol{n} - \boldsymbol{i}), \tag{1}$$

where $\hat{\boldsymbol{x}}$ denotes the signal predicted, $x(n - i)$ the previously observed value, and a_i the coefficients of the predictor. The error generated by this process is

$$\boldsymbol{e}(\boldsymbol{n}) = \boldsymbol{x}(\boldsymbol{n}) - \hat{x}(n), \tag{2}$$

where $x(n)$ is the signal value expected.

5 Result and Discussion

The analysis of Bodo monophthongs was done using MATLAB 7.1 and COLEA. Each recorded voice is first digitized and then is divided into 30 frames each of duration 18 ms (ms) each. Each frame contains approximately 386 samples, and for each frame, first (F1), second (F2), and third (F3) formant frequencies are computed and analyzed. The different formant frequencies for the Bodo vowels /i/ and /o/ associated with the selected informants are shown in Tables 3 and 4 for male and female, respectively.

From the current study on Bodo monophthongs articulated by female and male informants, it is found that for both male and female the third formant frequency

Table 3 Range of variation of formant frequencies of Bodo vowels (Female)

Vowel		F1 (kHz)	F2 (kHz)	F3 (kHz)
i	Max	2.370137	3.131033	3.998522
	Min	0.26401	2.398266	3.068157
	Average	0.424432	2.630018	3.679552
	Range	2.106127	0.732767	0.930365
o	Max	2.023202	3.780805	3.279643
	Min	0.325971	0.655054	2.473696
	Average	0.593461	1.322625	2.866332
	Range	1.697231	3.125751	0.805947

Table 4 Range of variation of formant frequencies of Bodo vowels (Male)

Vowel		F1 (kHz)	F2 (kHz)	F3 (kHz)
i	Max	0.338205	3.414299	3.988885
	Min	0.25402	2.701022	3.50481
	Average	0.294044	2.962312	3.734845
	Range	0.084185	0.713277	0.484075
o	Max	0.623019	1.189959	3.999092
	Min	0.282193	0.821004	3.28822
	Average	0.503701	0.898964	3.531814
	Range	0.340826	0.368955	0.710872

(F3) does not carry any remarkable characteristics which contradicts with the standard theory [3]. As a result, F3 may not be applicable for the purpose of speech as well as speaker recognition. On the other hand, the first formant (F1) variation across the vowels is quite distinct for both male and female speaker. So, F1 can be considered as a valuable cue for the purpose of speech and speaker recognition and identification. Although the result obtained for F2 is not so prominent, still it can also be considered for the speech and speaker recognition.

However, the role of F2 with respect to /a/, /e/, /o/, /w/ is very distinct for male and female. So, the values can be considered for the identification of gender for Bodo informants.

References

1. http://in.mathworks.com/help/signal/ug/formant-estimation-with-lpc-coefficients.htm
2. Rabiner, L.R., Schafer R.W.: Theory and Application of Digital Speech Processing, pp. 425–452 (2009)
3. Basumatary, P.: An Introduction to BORO Language, 1st edn. (2010)
4. Welling, L., Ney, H.: Formant estimation for speech recognition. IEEE Trans. Speech Audio Process. **6**, 36–48 (1998)
5. Sharma, U.: Formant Frequency Measure and Analysis of Bodo Vowels—A Core Language of NE India'. In: International Conference on Information & Mathematical Sciences (IMS-13), 24–26, ISBN: 978-93-5107-162-4, pp. 32–34. (Published by Elsevier) (Oct 2013)
6. Sharma, U.: Measurement of Formant Frequency for Consonant-Vowel (CV) Type Bodo Words for Acoustic Analysis. In: International Conference on Data Mining and Intelligent Computing (ICDMIC—2014). (Available in IEEE Explore) (5th–6th Sept 2014)
7. Reynolds, D. et al.: The Super SID project: exploiting high-level information for high-accuracy speaker recognition. In: Proceeding of International Conference on Acoustics, Speech, and Signal Processing (ICASSP 2003), pp. 784–787 (2003)
8. Stephenson, T.A., Doss, M.M., Bourlard, H.: Speech recognition with auxiliary information. Speech Audio Process. IEEE Trans. **12**(3), 189–203 (2004)
9. http://homepages.wmich.edu/~hillenbr/204.htm
10. Chandan Sarma, U.S., Nath, C.K., Kalita, S., Talukdar, P.H.: Selection of Units and Development of Speech Database for Natural Sounding Bodo TTS System, CISP Guwahati (Mar 2012)
11. Kalita, K., Sharma, U.: Extraction of Mel-Cepstrum from a Speech Signal with a Specific Reference to Bodo Phoneme. In: International Conference on Information & Mathematical Sciences (IMS-13), 24–26, ISBN: 978-93-5107-162-4, pp. 42–44. (Published by Elsevier) (Oct 2013)

Detection of Human Emotion from Speech—Tools and Techniques

Abhijit Mohanta and Uzzal Sharma

Abstract In the area of human–computer interaction (HCI), speech emotion recognition is an important topic. Various important research works on emotional speech analysis have been carried out in recent years. Different researchers have been introduced many systems to identify the emotion from human speech. This paper will give an idea about different techniques and working procedure of speech emotion recognition system. Also it gives the brief idea about the emotional speech dataset. We have reconsidered some earlier implemented speech emotion recognition technologies which use various feature extraction method and classifier for emotion recognition. Different types of classifier performance are also discussed for speech emotion recognition.

Keywords HCI · MFCC · LPCC · GMM · MFB · Prosody · Classifier

1 Introduction

Human emotion recognition through speech aims at automatically judging the physical or emotional state of human being from their speech signal. For decades, speech emotion recognition is a research hotspot in the field of human–computer interaction (HCI). From the ancient days, speech is considered as one of most primary as well as natural communication methods between human beings. Speech signal is one of the most efficient and fastest ways to recognize human emotion automatically. The human mind can effortlessly identify the emotional state but the same thing is quite hard task for machine to make out. The prime motive of

A. Mohanta (✉) · U. Sharma
Department of CSE & IT, School of Technology, Assam Don Bosco University,
Guwahati, India
e-mail: abhijit.mohanta@outlook.com

U. Sharma
e-mail: druzzalsharma@gmail.com

S. S. Agrawal et al. (eds.), *Speech and Language Processing for Human-Machine Communications*, Advances in Intelligent Systems and Computing 664,
https://doi.org/10.1007/978-981-10-6626-9_20

designing the emotion recognition system is to provide proper human–computer interaction by using emotion-related knowledge.

Speech emotion recognition system contains five elementary modules, i.e., emotional speech input, feature extraction, feature selection, classification, and output [1–3]. Feature vector is classified into four groups; namely special, continuous, qualitative and teager energy operator-based features. Also selection of feature vector is very much important [3]. The whole speech emotion recognition system may be speaker dependent and speaker independent. A system is said to be speaker dependent if it is developed to work for a specific speaker, and on the other hand, a system is said to be speaker independent if it is designed to work for any speaker. There are various classifiers available for emotion recognition. Some of these are Hidden Markov Model (HMM), Gaussian Mixture Model (GMM), Support Vector Machine (SVM), Artificial Neural Network (ANN), K-Nearest Neighbor (KNN) [1, 2].

Importance of emotion recognition system is that in case of the absence of a person the system can indentify his or her emotional state through speech. It is not necessary for the person to get face to face with the system. There are some barriers which increase the difficulties in order to get the more accurate output from emotional speech input. If it is not sure which speech features are need to be taken to distinguish between various emotional states then getting the exact output may be difficult. Speech features directly gets affected by speaker, speaking style, speaking rate, language, sentences. Changing of speaker and their environment and culture is also a big challenge in speech emotion recognition. With the changing culture and environment, the speaking style, speaking rate, etc., may also get changed [1, 2].

Pronunciation variance in emotional speech also matters in order to detect the underlying emotions in speech. There are many factors which create an effect on word pronunciation, for instance, the gender of the speaker, speaker age, word position within the utterance and dialect [4].

The application area of speech emotion recognition is very vast, few of its important applications are: psychiatric diagnosis, lie detection, intelligent toys, conversation with robots, identifying the emotional state of customer may help to enhance quality of service in call centers [1, 2, 5].

In this paper, we are going to discuss some fundamental things about the speech emotion recognition system. Sect. 2 of the paper consists of basic working procedure of speech emotion recognition system. Section 3 describes the categories of dataset. In Sect. 4, various feature extraction techniques have been discussed. Section 5 describes classifiers, Sect. 6 contains experimental study, and the paper concludes with Sect. 7.

2 Speech Emotion Recognition System

Speech emotion recognition system is a typical type of pattern recognition system which aim is to identify the emotional state of human beings automatically from his or her voice. Speech emotion recognition system consists of five main modules namely emotional speech input, feature extraction, feature selection, classification, and recognized emotional output.

A typical emotional dataset consists of 300 emotional states, so it is very difficult to identify such a huge number of emotions. But emotions are classified into 6 basic types namely fear, anger, happiness, sadness, neural, and surprise [1–3, 5] (Fig. 1).

3 Categories of Datasets

Dataset plays a very important role in speech emotion recognition system. The output of the speech emotion recognition system depends on the naturalness and efficiency of the input dataset and inexact result may occur because of the inappropriate dataset [1, 2]. Typically, there are three types of dataset used for the purpose of emotion recognition; these are actor-based emotional speech dataset, elicited emotional speech dataset, and natural emotional speech dataset. There is an international committee called COCOSDA for the standardization and coordination of speech dataset and assessment technique [6, 7].

In case of actor-based emotional speech dataset, data are collected by asking any trained actor or professional to speak with a specific type of emotion. This type of emotion is also known as full-blown emotion. There are many actor-based speech emotional dataset available but some are most commonly used and publicly available like Berlin Emotional Speech Dataset, Danish Emotional Speech Dataset, and Electromagnetic articulography (EMA) dataset. Elicited emotional speech dataset is collected by creating an artificial emotional situation, without speaker information. Since collecting this type of data is very much exhausting so only few numbers of elicited speech datasets are available. Natural emotional speech datasets are taken from real-life scenario like call center conversation, cockpit recordings, etc. [3, 8].

The databases that are nowadays being used, majority of them are actor-based emotional database. But nowadays a solid objection has been raised against the use of acted emotions because it found that acted samples are not same in terms of accuracy and feature. On the other hand, datasets of real emotional speech cause a

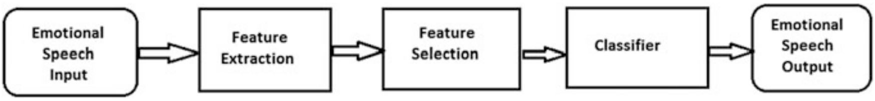


Fig. 1 Speech emotion recognition system

serious ethical problem. There should be the reveal of intimate and personal details about the speaker. The majority of emotional dataset encompasses five or six types of emotions, though there are much more categories exists in real life. But it is usually accepted that primary types of emotions are more primitive and universal compared to other types [3, 8, 6, 7].

In the present research, Bengali Emotional Speech is used as an experimental dataset. The recording of the speech signal was done in a sound proof room and using standard recording setup. Initially, a set of five speakers were considered for recording purpose.

4 Feature Extraction Techniques

Feature extraction deals with the analysis of speech signal. It depends on the partition of speech into small intervals called frame. Speech emotion contains a large number of parameters and the corresponding parameter gets changed based on the emotional state. In the field of speech emotion recognition, choice of appropriate feature vector is very important which is able to recognize the exact emotion type. Feature vector can be categorized into two types: long-time feature vector and short-time feature vector. Long-time feature vector is calculated based on the whole length of utterance, as opposed to short-time feature vector is calculated based on window which is generally less than 100 ms [1–3, 9, 10].

Linguistics prosody refers to the information pattern, speaking rate, speech rhythm, and character stress. Prosodic features are also called as the primary indicator of the speaker's emotional state. Emotional prosody features used to encode information at least from two sources, i.e., emotion and linguistics [11]. Research in the field of psycholinguistics shows that prosodic information like speaking rate and pitch is very much significant in human identification of underlying emotions in speech signal [4]. Linguistic information is a little portion of the spoken message which can be carried by text. Human minds are sensorial to paralinguistic information as well as extralinguistic information which are very much necessary but there is no use of this information in current speech recognition systems and in computer speech synthesis these informations are hugely missing. Paralinguistic and extralinguistic information model can be gained with respect to the semantic and structural utterance content [6].

Greater portion of the speech emotion research work has been concentrated on increasing the emotion classification efficiency. In spite of the comprehensive research in the field of emotion recognition, efficient speech normalization techniques have not been advanced yet, that can utilize the emotional state information in order to improve the speech recognition performance [7]. Emotion-specific acoustic and suprasegmental models can be used to identify the underlying state of emotion of the speaker with validity comparable to the human performance on the same task. We can significantly prosper the word accuracy of the speech recognition system by using emotion-specific modeling [4]. Every approach to the speech emotion recognition practically avoids the spoken content in case of acoustic modeling [12]. According to a hypothesis, word location also matters in case of emotional speech processing and its result indicates that emotional effects are usually weightier on sentence medial words rather than the initial and final words of sentence [13].

It is neatly understandable from the speech emotion recognition research that feature vectors like energy, pitch, intensity, format, speech rate, duration, mel-filter bank (MFB), mel-frequency cepstrum coefficient (MFCC), and Linear prediction cepstrum coefficient (LPCC) are very much significant in order to identify emotional state. Energy, pitch, speak rate, and spectrum are different in different emotional state. Basically, anger got the variance of pitch, high-frequency formats, mean value of energy, and higher mean value. On the other hand, happy state has an improvement in variance of pitch, variation range, mean value, and mean value of energy. Sadness has low mean value, variation range and variance of pitch, energy is weak, speak rate is slow and spectrum of high-frequency components reduces. Fear has a high mean value, higher energy, and high variation range of pitch. So the statics of pitch, energy, formats and some important spectrum feature can be extracted in order to identify the emotion from speech signal.

The extraction of all basic speech features for emotion recognition may not be necessary. After the extraction of feature vectors when we give the entire extracted feature as input to the classifier, it gives no guarantee that the system will give exact performance. It is very much important to extract a significant feature vector which is able to give large emotional information about the speech signal. Forward selection (FS) method could be used for selecting the significant feature subset. In the first step of the forward selection method, it initializes with the single best feature out of the whole features and for classification validity the other feature can be added in future [1–3].

5 Classifier

Appropriate feature extraction and classifier selection is very important in order to get the exact emotional output. So after selecting the feature vector, the most important task is to select the proper classifier. No fixed standard is there for classifier selection; it depends on the geometry of the input vector. There are different types of classifier for speech emotion recognition system. Gaussian Mixtures Model (GMM), Support Vector Machine (SVM), Hidden Markov Model (HMM), K-Nearest Neighbors (KNN), Artificial Neural Network (ANN), etc., are some of the widely used classifiers in emotion recognition system. Each classifier has its own advantages and disadvantages over others. Human mind could hardly recognize emotion in speech up to 60% in case of unknown speaker whereas advance researches got the accuracy rates from 55 to 99% in speaker-independent speech [1–3].

In the field of emotion recognition according to the various researchers, the accuracy rate of HMM in case of speaker-dependent classification is 76.12% and in case of speaker-independent classification, it is 64.77%. On the other hand, the accuracy rate of GMM in case of speaker-dependent and speaker-independent classification is, respectively, 89.12 and 75% [1–3, 14]. ANN classifier accuracy is comparatively lower than other classifiers, i.e., 52.87% in case of speaker-independent classification and in case of speaker-dependent classification accuracy rate is 51.19% [1–3, 5]. KNN has the accuracy rate of 64% for four emotional states by using the feature vector like energy contours, pitch. [1, 2].

6 Experimental Study

For our current research, we have taken SVM classifier. SVM is a type of binary classifier which is also used as a multiclass classifier. The working policy of SVM is to passing the original feature set to a higher dimensional feature space by using kernel function. SVM is able to classified emotional states into a huge margin. Margin refers the largest tube width without any utterances. Because of its structural risk minimization-oriented training, SVM has a good generalization capability [1–3, 5, 15]. To develop SVM models for a particular emotion, feature vectors those who are evolved from the desired emotional speech are used as positive examples, and the feature vector evolved from any other emotional speech are considered as negative examples [16] (Fig. 2).

We will be able to use the classifier to identify different emotional state, after having a set of features with us. At first, the classifier training has to done using some different emotional state input. Once the training has done, we can use the classifier to identify the new given input. In the SVM training process, each extracted feature must assign an associated class level and then the SVM training has to be done according to this labeled feature. By properly utilizing the above features, we can improve the result.

The feature vector which we have extracted is MFCC. For each feature vector, there must be a corresponding label of belonging class. Being a binary classifier, SVM classes are labeled as {+1, −1}. For non-separable data, we have implemented a SVM using linear kernel function. The test results are calculated with the help of multiclass SVM as well as our implemented SVM. Results are taken in case of multiclass SVM by using MFCC feature vector for all kernel functions. By using multiclass SVM, overall accuracy percentage is gained [17].

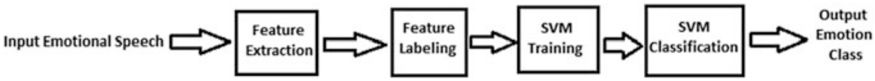


Fig. 2 Structure of speech emotion recognition system using SVM

Table 1 SVM Classifier Confusion Matrix for speaker-independent emotional speech

Emotion	Anger	Sad
Anger	63.63	4.93
Sad	4.76	75.12

Table 1 shows the implemented SVM confusion matrix of our study on Bengali emotional speech utterance by using one-to-one multiclass method. Sad emotion offers better recognition rate compared to angry. We have observed that in case of speaker-independent classification SVM has got the accuracy rate 75%. Now based on the above discussion if we consider the accuracy rate of speaker-independent system then we can say that GMM as well as SVM will give the best accuracy rate, i.e., 75% both.

7 Conclusion

To increase the human–computer interaction, the need of automatic human emotion recognition is increasing day by day [18, 19]. In this paper, we have reviewed some important emotion recognition technique, few commonly used feature extraction method and classifiers. Accuracy and efficiency of emotion recognition system depends on the appropriate feature extraction and classifier selection. We have observed that in most of the classifiers the average accuracy is more in case of speaker-dependent system compared to the speaker-independent system. So it is required to improve the classifier performance in case of speaker-independent classification. Extraction of more effective speech features can result a speech emotion recognition system with higher accuracy. More effective feature extraction can give a higher accuracy rate in speech emotion recognition system. Also the combination of various methods will improve the accuracy rate.

References

1. Utane, A.S., Nalbalwar, S.L.: Emotion recognition through speech. Int. J. Appl. Inf. Syst. (IJAIS) 5–8 (2013)
2. Mohanta, A., Sharma, U.: Human emotion recognition through speech. Adv. Comput. Sci. Inf. Technol. (ACSIT) **2**(10), 29–32 (Apr–June 2015)
3. Joshi, D.D., Zalte, M.B.: Speech emotion recognition: a review. IOSR J. Electron. Commun. Eng. (IOSR-JECE) 34–37, (Jan–Feb 2013)
4. Polzin, T.S., Waibel, A.: Pronunciation variations in emotional speech
5. Suri, P., Singh, B.: Enhanced HMM speech emotion recognition using SVM and neural classifier. Int. J. Comp. Appl. (0975–8887) 17–20, (Feb 2014)
6. Campbell, N.: Databases of emotional speech. In: ITRW on Speech and Emotion Newcastle, Northern Ireland, UK, September 5–7 (2000)

7. Ververidis, D., Kotropoulos, C.: Emotional speech recognition: resources, features, and methods, pp. 1–22 (Apr 19 2006)
8. Milton, A., Roy, S.S., Selvi, S.T.: SVM scheme for speech emotion recognition using MFCC feature. Int. J. Comp. Appl. (0975–8887) 34–39, (May 2013)
9. Han, K., Yu, D., Tashev, I.: Speech emotion recognition using deep neural network and extreme learning machine. In: Interspeech 2014, pp. 223–227, (Sept 2014)
10. Joshi, A.: Speech emotion recognition using combined features of HMM & SVM algorithm. Int. J. Adv. Res. Comput. Sci. Softw. Eng. 387–393, (Aug 2013)
11. Jiang, D., Zhang, W., Shen, L., Cai, L.: Prosody analysis and modeling for emotional speech synthesis. pp. 281–284
12. Schuller, B., Vlasenko, B., Arsic, D., Rigoll, G., Wendemuth, A.: Combining speech recognition and acoustic word emotion models for robust text-independent emotion recognition, pp. 1333–1336
13. Kim, J., Lee, S., Narayanan, S.S.: A detailed study of word-position effects on emotion expression in speech
14. Utane, S.A., Nalbalwar, S.L.: Emotion recognition through speech using gaussian mixture model and hidden markov model. Int. J. Adv. Res. Comput. Sci. Softw. Eng. 742–746, (Apr 2013)
15. Chavhan, Y., Dhore, M.L., Yesaware, P.: Speech emotion recognition using support vector machine. Int. J. Comp. Appl. (0975–8887), 6–9
16. Koolagudi, S.G., Kumar, N., Rao, K.S.: Speech emotion recognition using segmental level prosodic analysis. IEEE (2011)
17. Chavan, V.M., Gohokar, V.V.: Speech emotion recognition by using SVM-classifier. Int. J. Eng. Adv. Technol. (IJEAT) **1**, 11–15, (June 2012)
18. Sharma, S., Singh, P.: Speech emotion recognition using GFCC and BPNN. Int. J. Eng. Trends Technol. (IJETT) 321–322, (Dec 2014)
19. Pan, Y., Shen, P., Shen, L.: Feature extraction and selection in speech emotion recognition. pp. 64–69

Phonetic Transcription Comparison for Emotional Database for Speech Synthesis

Mukta Gahlawat, Amita Malik and Poonam Bansal

Abstract Phonetics transcription is the process of representing the speech unit into phonetic alphabets. This is necessary step for doing speech synthesis. It involves segmentation and labelling of sound files. Transcription at phonetic level can be performed either manually or automatically. Both ways are implemented on different expressions like happy, neutral and sad. Comparisons using various parameters like pitch, power and formants are made for various emotions. Additionally, pros and cons of using manual and automatic segmentations are also discussed on the basis of result received on expressive speech corpus.

Keywords Speech synthesis · Segmentation · Emotional database

1 Introduction

Speech synthesis is one of the major sub-fields under speech technology. It involves conversion of given text into speech. Research on speech synthesis is going on since several decades. But still there are scopes of improvement in the quality of speech. Naturalness and intelligibility are the two most important parameters to determine the performance of Text To Speech Synthesizer (TTS). To develop the natural speech, developer relies on concatenative corpus based speech synthesis, which in turn depends on accessibility of high-quality speech database. In order to achieve this objective, accurate segmentation and labelling of speech units are required. Phonetic transcription is the crucial step that is directly proportional to

M. Gahlawat (✉) · A. Malik
Computer Science & Engineering Department, DCRUST, Murthal, India
e-mail: mukta.gahlawat@gmail.com

A. Malik
e-mail: amitamalik.cse@dcrustm.org

P. Bansal
Computer Science & Engineering Department, MSIT, C-4 Janakpuri, Delhi, India
e-mail: pbansal89@yahoo.co.in

S. S. Agrawal et al. (eds.), *Speech and Language Processing for Human-Machine Communications*, Advances in Intelligent Systems and Computing 664,
https://doi.org/10.1007/978-981-10-6626-9_21

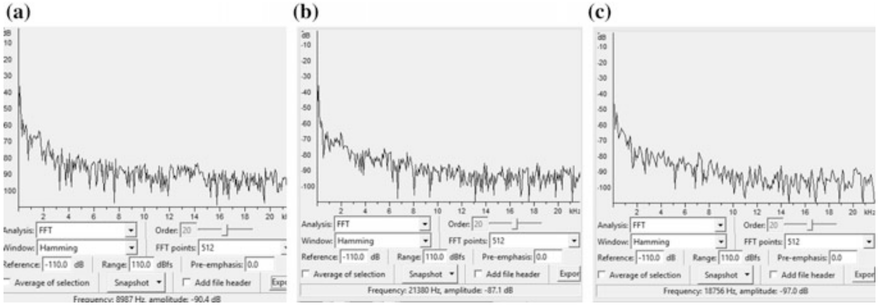
output of speech. But at the same time it is well-known fact that it consumes most of the time and effort during corpus creation. There are two ways of doing these manual and automatic transcriptions. Both are discussed in this paper on emotional corpus.

2 Related Work

Annotation of speech is one of the mandatory steps that are required for conversion of written text into sound. Research has been done for efficient segmentation of speech units. Knut Kvale [1] has worked on segmentation and labelling. He defined "segmentation as the process of dividing the speech waveform into directly succeeding discrete parts". After segmentation, the phoneme symbols are labelled with these segments accordingly. Acoustic and phonemic segmentation algorithms were proposed for doing automatic segmentation. Doroteo Torre Toledano et al. [2] perform phonetic segmentation automatically using HMM. As automatic segmentation is less precise than manual segmentation, so to increase the performance a new system design was purposed. The phonetic transcription based on this framework yields good results. But good and standard way for evaluation of transcription is also required. Maria-Barbara Wesenick [3] provide a way to evaluate segmentation. Their algorithm is based on pattern matching that will be done automatically. Using this segmentation based on automatic means, automatic and manual are compared with each other. On the basis of matching, their identification degree is calculated. As a result, final outcome can also give a view about the quality of transcription and segmentation. It has been found in some work that using modified algorithm for automatic segmentation has increased the Text To Speech Synthesizer performance [4]. Further, some processes were employed to enhance the efficiency in segmentation, for instance, application of zero crossing to all the phonemes [5]. The automation of phonetic segmentation in Spanish has been attempted on various emotions [6].

3 Comparative Analysis of Different Emotions

Firstly, the understanding of various expressions for speech synthesis is required to perform phonetic transcription. Various types of expressions can be categorized on the basis of some parameters. There are three parameters that are considered for their comparison. These are pitch, formats and power. Figure 1 shows the spectrum for sentence "During the Summer" happy, neutral and sad. Figure 2 shows pitch, power, formant, waveform in happy expression for sentence "During the Summer". Figure 3 shows pitch, power, formant and waveform in sad expression for sentence "During the Summer". Figure 4 shows pitch, power, formant, waveform in neutral expression for sentence "During the Summer". After studying the waveforms, it has



(a)

Analysis: FFT Order: 20
Window: Hamming FFT points: 512
Reference: -110.0 dB Range: 110.0 dBfs Pre-emphasis: 0.0
□ Average of selection □ Add file header
Frequency: 8987 Hz, amplitude: -90.4 dB

(b)

Analysis: FFT Order: 20
Window: Hamming FFT points: 512
Reference: -110.0 dB Range: 110.0 dBfs Pre-emphasis: 0.0
□ Average of selection □ Add file header
Frequency: 21380 Hz, amplitude: -87.1 dB

(c)

Analysis: FFT Order: 20
Window: Hamming FFT points: 512
Reference: -110.0 dB Range: 110.0 dBfs Pre-emphasis: 0.0
□ Average of selection □ Add file header
Frequency: 18756 Hz, amplitude: -97.0 dB

Fig. 1 Showing spectrum for sentence "During the Summer" **a** happy, **b** neutral, **c** sad

been found that pitch is almost same for sentences spoken in neutral but significant variations have been found for sad and happy emotions. The power is highest for happy emotion, but when the same sentence spoken in sad and neutral the power goes up to 20 db. Variations for more in sad and happy emotions.

4 Phonetic Transcription

For developing natural speech, expressions play an important role. There are two ways for segmentation, i.e. manual and automatic. Table 1 shows comparisons between these two on emotional database.

4.1 Manual Segmentation

The degree of naturalness in synthesized speech also depends upon the way of generation and type of speech corpora used. The corpora available online are not fulfilling our requirements. The requirements such as domain, accent, language, unit, size and platform are not meeting in any corpora available. So, personalized speech corpora are taken. The main challenge in development in speech corpora is its labelling and segmentation as huge amount of time and effort is required for its annotation. The database was initially formed using manual segmentation using wavesurfer [8].

5 Automatic Segmentation

Lots of time and efforts are required for doing manual segmentation. Another cause of shifting to automatic segmentation is requirement of extension of database, which is continuous process. As the size of database increases, the time and effort required for segmentation increases tremendously. So to reduce this immense effort,

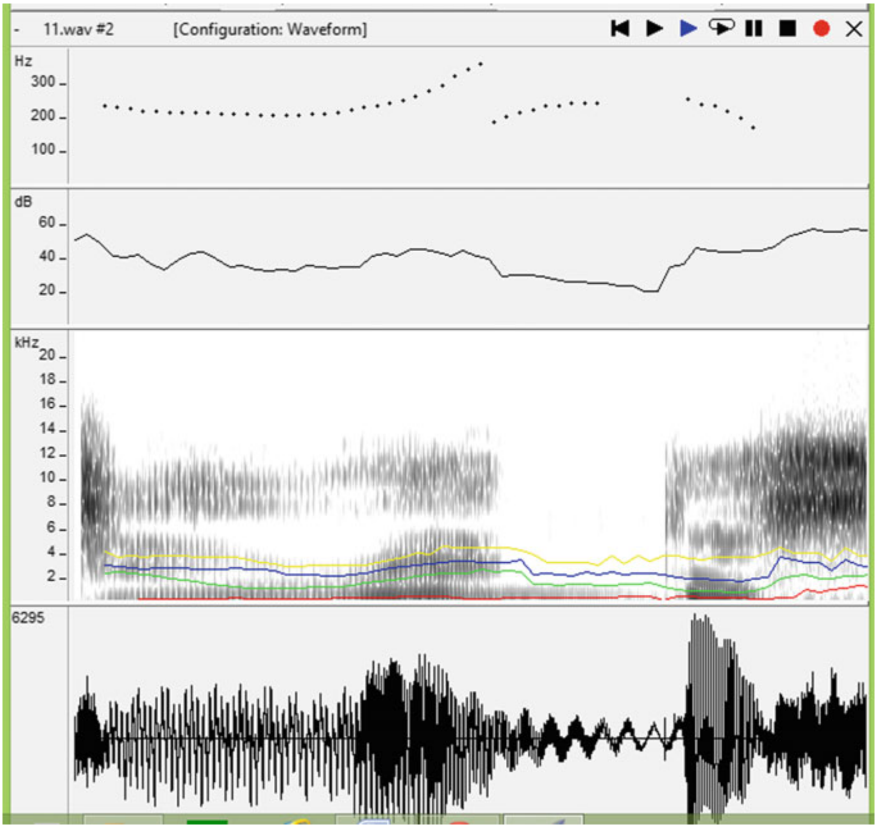


Fig. 2 Pitch, power, formant, waveform in happy expression for sentence "During the Summer"

automatic segmentation is carried out. Automatic segmentation is done on the speech wave files. This automatic segmentation has significantly reduced manual effort and time incurred in annotation of speech database. Automatic segmentation means conversion of audio file into annotated small speech units like phoneme or syllables. It is done using Hidden Markov Model Toolkit [9, 10] that was developed at Cambridge University Engineering Department. It provides various tools for analysis of speech, training of HMM, testing and analysis of result. Input taken is audio file and its transcription at sentence level. After providing input, there are number of operations that are requisite in order to get the final outcome. For illustration, output segments of one sentence (once upon a time) are shown below in Tables 2 and 3. As a result, the average percentage performance is 23.17 and root-mean-square percentage performance is 36.55. So to improve the performance, the manual cross-checking of segmentations is performed. Manual segmentation is more precise but is time consuming. So, semi-automatic approach has been undertaken for more precise boundary labelling.

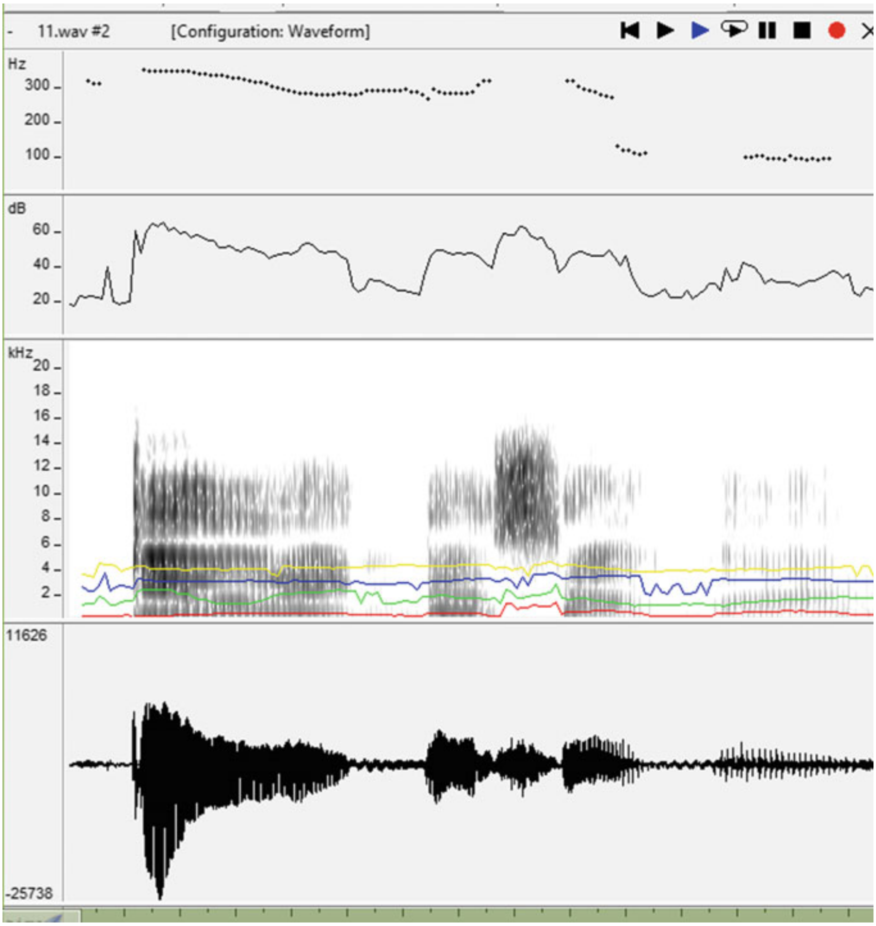


Fig. 3 Pitch, power, formant, waveform in sad expression for sentence "During the Summer"

6 Conclusion

The objective of this paper to understand the process of transcription on expressive corpora using manual and automatic segmentations has been achieved. Further, the comparative analysis of sound waveforms using different expressions is carried out on the basis of parameters like pitch, power and formants. This analysis can be useful at the places where the conversion of neutral speech to expressive speech is required by changing the speech prosody. It has been seen that embedding right

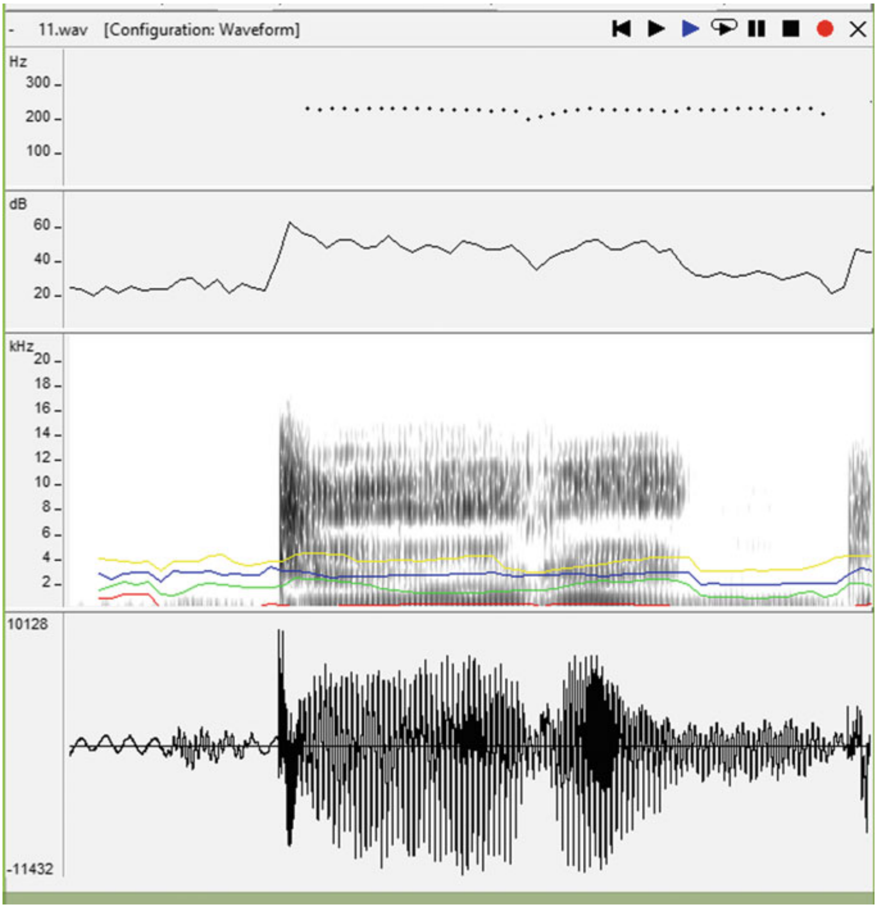


Fig. 4 Pitch, power, formant, waveform in neutral expression for sentence “During the Summer”

expressions in the synthesized speech increases the intelligibility and naturalness. Hence, it will be interesting to explore the phonetic transcription of expressive speech for different emotions. Automatic segmentation reduces the time and effort but the manual segmentation provides more precise boundaries for emotional corpus too. The automatic segmentation used on emotional corpus was corrected using manual efforts.

Table 1 Comparison of three emotions in manual and automatic segmentations

Phonetic transcription	Happy		Neutral		Sad		Automatic segmentation
	H1	H2	N1	N2	S1	S2	
_	0	619	0	7236	0	4851	Sil
D	619	3369	7236	9758	4851	9758	D
U	3369	7082	9758	15,335	9758	13,230	Y
R	7082	11,139	15,335	18,521	13,230	16,648	Ua
I	11,139	14,233	18,521	21,840	16,648	18,743	R
N	14,233	18,152	21,840	25,558	18,743	21,499	Ih
G	18,152	20,283	25,558	27,881	21,499	23,373	Ng
D	20,283	21,796	27,881	29,673	23,373	29,673	Sil
@	21,796	23,515	29,673	31,067	29,673	33,296	Dh
S	23,515	28,947	31,067	36,245	33,296	39,029	Ax
@	28,947	34,103	36,245	40,428	39,029	43,218	S
M	34,103	39,191	40,428	45,340	43,218	47,849	Ah
@	39,191	43,660	45,340	51,248	47,849	55,235	M
R	43,660	47,786	51,248	60,542	55,235	65,489	Ax
_	47,786	60,437	60,542	64,857	65,489	72,765	Sil

Table 2 Manual segmentation

0	7236	–
7236	27,881	During
27,881	31,067	The
31,067	60,542	Summer
60,542	64,857	Sil

Table 3 Automatic segmentation

0	0	Sil
0	14,300,000	During
14,300,000	18,600,000	The
18,600,000	40,400,000	Summer
40,400,000	40,400,000	Sil

References

1. Kvale, K.: Segmentation and labelling of speech PhD Thesis submitted at Department of Telecommunication of The Norwegian Institute of Technology (1993)
2. Toledano, D.T., Luis, A., Gómez, H., Grande, L.V.: Automatic phonetic segmentation. IEEE Trans. Speech Audio Process. **11**(6) (Nov 2003)
3. Wesenick, M.B., Kipp, A.: Estimating the Quality of Phonetic Transcriptions and Segmentations of Speech Signals, these Proceedings of ICSLP, Philadelphia/ USA (1996)
4. Sudhakar, B., Raj, R.B.: Automatic speech segmentation to improve speech synthesis performance. In: 2013 International Conference on Circuits, Power and Computing Technologies [ICCPCT-2013], pp. 835–839

5. Szklanny, K., Wójtowski, M.: Automatic segmentation quality improvement for realization of unit selection speech synthesis. In: IEEE Conference HSI, pp. 251–256 (2008)
6. Gallardo-Antolín, A., Barra-Chicote, R., Schröder, M., Krstulovic, S., Montero, J.M.: Automatic phonetic segmentation of Spanish emotional speech. In: Interspeech, pp. 2905–2908. ISCA (2007)
7. Audacity, http://audacity.sourceforge.net/manual-1.2/tutorials.html
8. Wavesurfer, http://www.speech.kth.se/wavesurfer/
9. Young, S., Odell, J., Ollason, D., Valtchev, V., Woodland, P.: The HTK Book, , Version 2.1. Cambridge University (1997)
10. Salvi, G.: HTK Tutorial. K.T.H. Royal Institute of Technology, Department of Speech, Music and Hearing, Drottning Kristinas v. 31, SE-100 44, Stockholm, Sweden

The State of the Art of Feature Extraction Techniques in Speech Recognition

Divya Gupta, Poonam Bansal and Kavita Choudhary

Abstract This paper surveys feature extraction techniques applied in automatic speech recognition. After so many researches and improvement, the accuracy is a key issue in speech recognition systems. Speech recognition process converts the speech signal into its corresponding written text by the computer system. In this paper, we brief few well-known techniques of feature extraction like LPC, MFCC, RASTA, PCA, LDA, PLP.

Keywords Feature extraction · LPC · MFCC · RASTA · PCA · LDA
Automatic speech recognition

1 Introduction

Speech recognition technology has allowed humans to communicate with machines using voice commands and instructions. Thus, this technology is adopted for many applications in current time including cellular systems, telephone, and other areas [1]. The key reason behind performance degradation of speech recognition systems is the mismatch problem that arises due to the discrepancy between the testing and application environments which has been contaminated with noise. Speech recognition process transforms the acoustic signals into stream of words [2]. The performance of these systems is greatly influenced by several factors including surroundings, vocabulary, speaker variability, etc. Speech recognition systems perform well in clean environment with small vocabulary having few utterances for

D. Gupta
Computer Science Department, Amity University Uttar Pradesh, Noida, India

P. Bansal (✉)
Computer Science Department, GGSIPU, Dwarka, Delhi, India
e-mail: pbansal89@yahoo.co.in

K. Choudhary
Computer Science Department, Jagannath University, Jaipur, India

S. S. Agrawal et al. (eds.), *Speech and Language Processing for Human-Machine Communications*, Advances in Intelligent Systems and Computing 664,
https://doi.org/10.1007/978-981-10-6626-9_22

a given speech. The performance of the system is decreased in the presence of noise.

All speech recognition systems include two major stages that greatly influence the working and recognition rate of the system. One is the front-end stage that converts speech samples into stream of feature vectors coefficients which contains only that information which is required for the identification of a given utterance [3, 4]. There are various methods for feature extraction like LPC, MFCC, RASTA. [5]. The other stage is the classification or pattern matching which determines the category for each pattern like SVM, DTW, and HMM. [6].

The paper is framed as follows: Section 2 gives the description of speech recognition techniques. Section 3 describes different feature extraction techniques that are widely used along with their characteristics, advantages, and disadvantages. Section 4 presents the performance comparison among various automatic speech recognition systems.

2 Speech Recognition Techniques

The key point in speech recognition systems is to hear, understand and then working on spoken information. ASR system broadly classified as in Fig. 1 [7].

2.1 Analysis

Analysis is the first stage of ASR system. To show the speaker identity, speech data contains the speaking tract, the source of excitation, and the behavior feature. Speech analysis stage is broadly classified in three stages: analysis of segments, analysis of subsegmental, and analysis of suprasegmental.

2.2 Method of Feature Extraction

The main phase of an ASR system is feature extraction [8]. It plays very important role in the system. Feature extraction helps the system in identifying the speaker by extracting features from input signal.

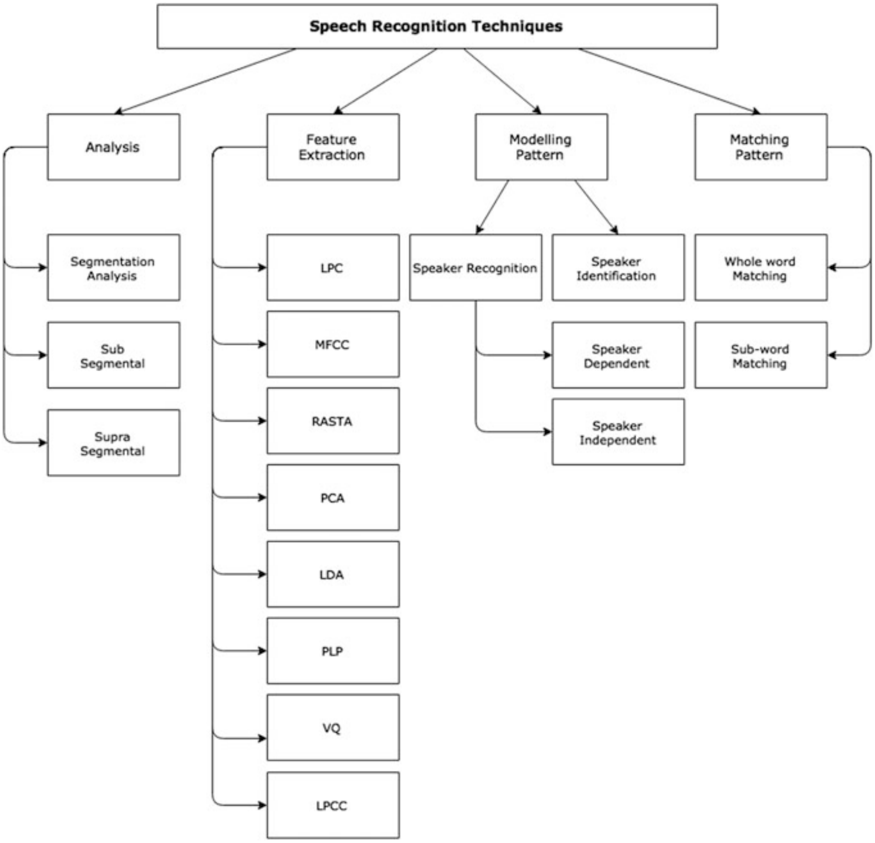


Fig. 1 Speech recognition techniques [14]

2.3 Modeling Pattern

Speaker identification and speaker recognition are the two modeling techniques that are used in ASR systems [4, 9]. Speech signal extracts the information, which helps in identifying the speaker. Acoustic-phonetic approach and dynamic time warping (DTW) are few common modeling approaches in speech recognition process.

2.4 Matching Pattern

This technique focuses on the recognition of words. The recognized word is used by speech recognition engine and after that it matches to a word that is already known [7, 10]. This technique is performing by either using sub-word matching or whole word matching method.

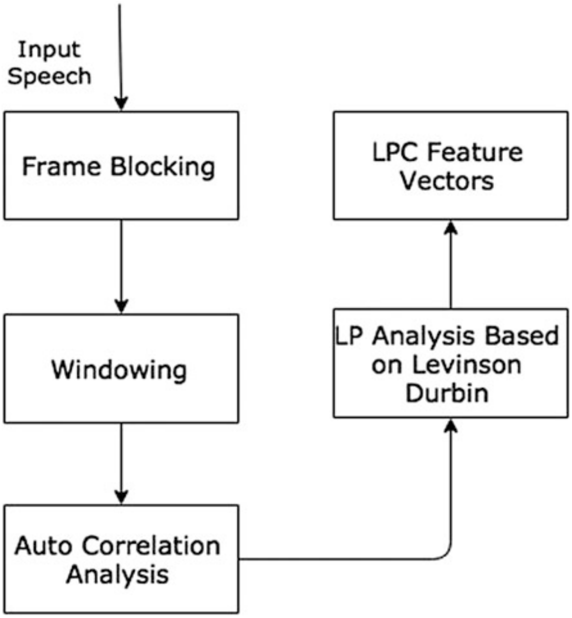
3 Techniques Used for Feature Extraction

The following are the few techniques used in feature extraction method.

3.1 Linear Predictive Coding (LPC)

LPC technique mainly performs the speech processing, and it is based on an assumption concept. By taking the bunch of speech samples, we can easily assume the nth sample. The basic idea of linear prediction is that the current speech sample can be closely approximated as a linear combination of past samples [11] (Fig. 2).

Fig. 2 LPC feature extraction technique


Input Speech
Frame Blocking
Windowing
Auto Correlation Analysis
LP Analysis Based on Levinson Durbin
LPC Feature Vectors

Technique	Characteristics	Advantages	Disadvantages
Linear predictive coding (LPC)	• Provides auto-regression-based speech features • Is a formant estimation technique • A static technique • The residual sound is very close to the vocal tract input signal	• Is a reliable, accurate, and robust technique for providing parameters which describe the time-varying linear system which represent the vocal tract • Computation speed of LPC is good and provides with accurate parameters of speech [12]	• Poor speech quality and it gives residual error as output • Is not able to distinguish the words with similar vowel sounds [3] • Cannot represent speech because of the assumption that signals are stationary and hence it is not able to analyze the local events accurately

3.2 *Mel-Frequency Cepstrum (MFFC)*

The cepstral coefficients are extracted from speech signals in twofold stages. The mel-scale filter bank is a technique for spectral estimation. It determines narrow-band filter energies. Next, cepstral analysis stage of processing codes the filter energies by using a Fourier transform. A mel-scale filter bank is array of covering triangular filter with center occurrences and bandwidths determined by the Mel-frequency scale. It is based on results from psychophysical learning of humans. MFCC is an eminent techniques used in speaker recognition which is focused on the speaker discriminative vocal tract properties (Fig. 3).

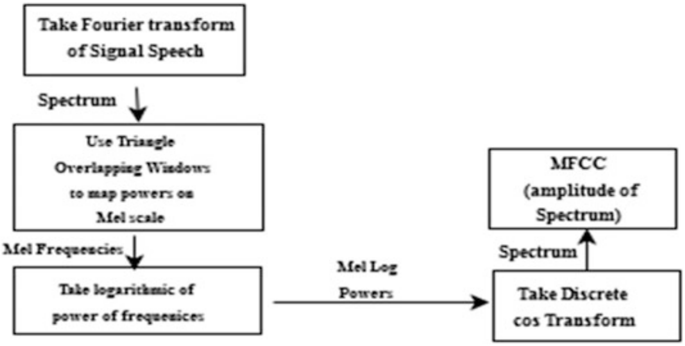

Take Fourier transform of Signal Speech
Spectrum
Use Triangle Overlapping Windows to map powers on Mel scale
Mel Frequencies
Take logarithmic of power of frequenices
Mel Log Powers
Take Discrete cos Transform
Spectrum
MFCC (amplitude of Spectrum)

Fig. 3 MFCC feature extraction technique

Technique	Characteristics	Advantages	Disadvantages
Mel-frequency cepstrum (MFCC)	• Used for speech processing tasks [13] • Mimics the human auditory system [14]	• MFCC captures main characteristics of phones in speech • The recognition accuracy is high. That means the performance rate is high • Low complexity [12]	• The filter bandwidth is not an independent design parameter • In background noise, MFCC does not give accurate results [4]

3.3 Relative Spectral (RASTA)

In noisy environment, to enhance the speech quality, RASTA technique is very useful. In RASTA, the time trajectories in the input speech signals are band-pass filtered [15, 16]. The step-by-step working of RASTA is shown in the following Fig. 4.

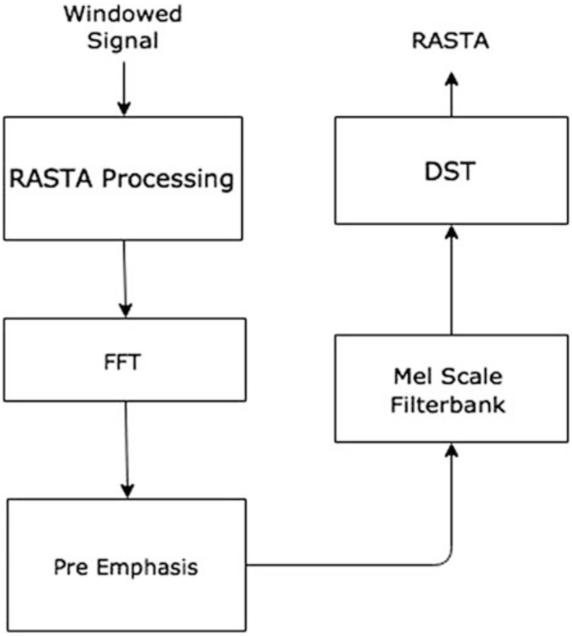


Fig. 4 RASTA feature extraction technique

Technique	Characteristics	Advantages	Disadvantages
Relative spectral (RASTA filtering)	• Designed to lessen impact of noise as well as enhance speech. That is, it is a technique which is widely used for the speech signals that have background noise or simply noisy speech • Is a band-pass filtering technique	• This technique does not depend on the choice of microphone or the position of the microphone to the mouth, hence it is robust [13, 17] • Captures frequencies with low modulations that correspond to speech • Removes the slow varying environmental variations as well as the fast variations in artifacts	• This technique causes a minor deprivation in performance for the clean information, but it also slashes the error in half for the filtered case. RASTA combined with PLP gives a better performance ratio

3.4 Principal Component Analysis (PCA)

PCA technique is used in the reduction of high-dimensional data into smaller dimensions by considering different characteristics [16]. The step-by-step processing in PCA is shown in Fig. 5.

Technique	Characteristics	Advantages	Disadvantages
Principal component analysis (PCA)	• PCA does not deal with the classification feature • While transformed to a different space than the structure and location change	• Robust in nature [4] • Retain more significant information and decrease in the feature vector's size [9]	• For high-dimension data, PCA is expensive [8]

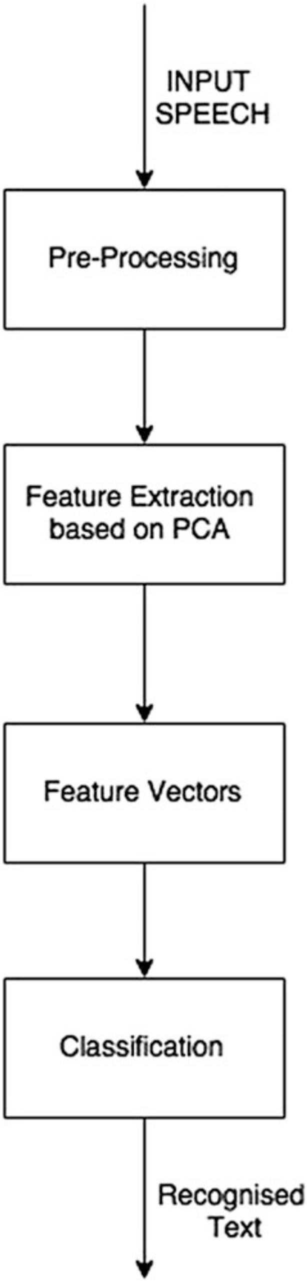


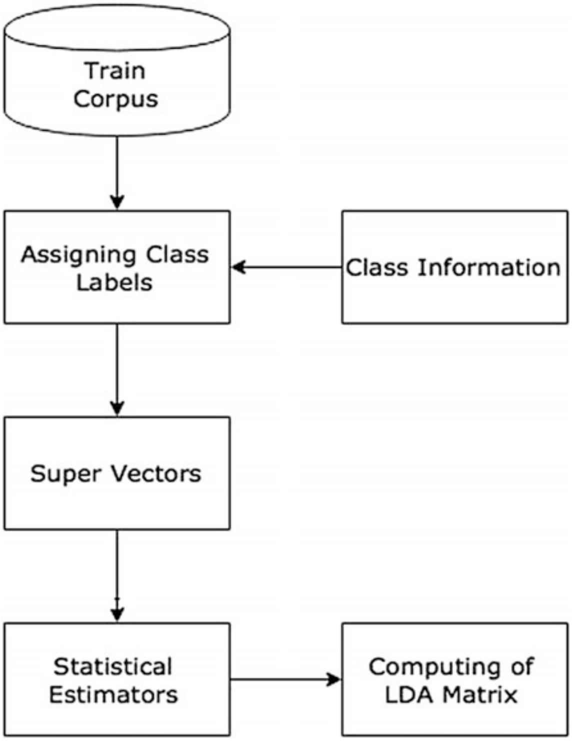
Fig. 5 PCA feature extraction technique

3.5 Linear Discriminant Analysis (LDA)

In LDA technique, the original feature does not change the location or the structure [9]. LDA works in two steps as shown in Fig. 6.

Technique	Characteristics	Advantages	Disadvantages
Linear discriminant analysis (LDA)	• The location or the structure of the original features does not change [18] • Deals with data classification [19]	• Robust in nature • Within the class, distance is reduced and increases the distance between classes [4]	• Sample distribution is assumed on priority to be Gaussian [3] • It assumes that class samples have equal variance

Fig. 6 LDA feature extraction technique



3.6 Perceptual Linear Predictive Cepstrum (PLP)

PLP is used to emphasize the need for critical band analysis that merges the energy spectral density for obtaining the speech auditory spectrum. The techniques for calculating the LP cepstral coefficients are same with the method for figuring the PLP factors [20] (Fig. 7).

Technique	Characteristics	Advantages	Disadvantages
PLP	• Similar to LPC except, the spectral characteristics • Unwanted information of speech has been discarded	• Low-dimensional resultant feature vector • Difference between voiced and unvoiced speech is reduced • It is used in speech signal that is based on short-term spectrum [21]	• Communication channel, noise the spectral balance is easily changing [22] • In the spectral balance of the format amplitudes, the result feature vectors are dependent

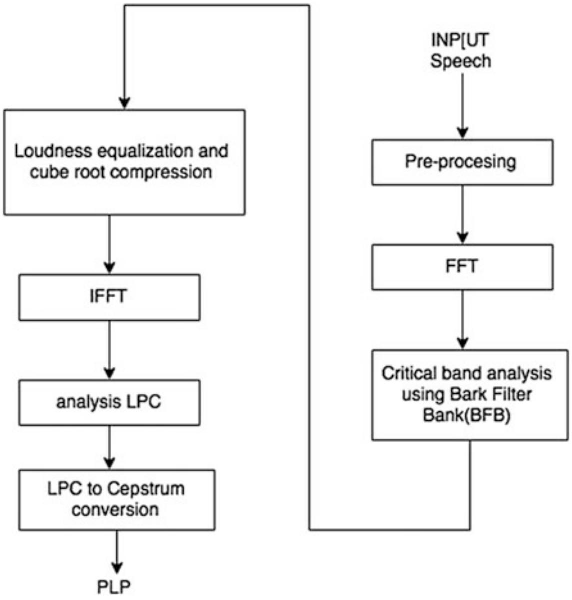

INP[UT
Speech
Pre-procesing
FFT
Critical band analysis using Bark Filter Bank(BFB)
Loudness equalization and cube root compression
IFFT
analysis LPC
LPC to Cepstrum conversion
PLP

Fig. 7 PLP feature extraction technique

Table 1 Performance contrast between various ASR systems

Year/reference	Features extraction technique	Feature classification technique	Speaker dependent/ speaker independent	Accuracy
2014 [19]	RASTA-MFCC	GMM-UBM	SD	93.4%
2014 [15]	RASTA-MFCC	UBM-SVM	SI	MFCC-67.6, RASTA-70.5
2009 [23]	MFCC	SVM	SI	94.35%
2010 [16]	MFCC PLP PCA	SVM	SI	HMM—70.42% SVM—71.75%
2012 [18]	MFCC PITCH	GMM	SI	79.9% (female) 89.02% (male)
2013 [24]	Energy ZCR MFCC	SVM	SI	89.8%
2011 [21]	LPCC MFCC 89.27	Modified-SOM	SI	88.05
2007 [25]	MFCC	Euclidean distance measure, vector quantization (VQ)	SI	88.8%

4 Summary of Automatic Speech Recognition Systems

The following Table 1 shows the performance comparison among various automatic speech recognition systems.

5 Conclusion

In this paper, we summarized some of the feature extraction techniques which are mainly used in the area of automatic speech recognition. The main objective of this review paper is to give a brief overview of different feature extraction techniques. We attempt to provide a comprehensive survey of six feature extraction techniques which help to researchers in the field of automatic speech recognition area. We have also summarized the performance comparison of various ASR systems.

References

1. Bhabad, S.S., Kharate, G.K.: An overview of technical progress in speech recognition. Int. J. Adv. Res. Comput. Sci. Soft. Eng. **3**(3) (2013)
2. Nehel, N.S., Holambe, R.S.: DWT and LPC based feature extraction methods for isolated word recognition. J. Audio Speech Music Process (2012)
3. Mishra A.N., Shrotriya, M.C., Sharan, S.N.: Comparative wavelet, PLP and LPC speech recognition techniques on the Hindi speech digits database. ICDIP Singapore (2010)
4. Zhang, G., Song Q., Fei, S.: Research on speech emotion. Comput. Technol. Prospect **19**, 92–95 (2009) (in Chinese)

5. Wijoyo, T.S.: Speech recognition using linear predictive coding and artificial neural network for controlling movement of mobile robot. In: International Conference on Information and Electronics Engineering
6. Tiwari, V.: MFCC and its applications in speaker recognition. Int. J. Emerg. Technol. **1**(1), 19–22(2010). National Center for Biotechnology Information, http://www.ncbi.nlm.nih.gov
7. Anusuya, M.A., Katti, S.K.: Speech recognition by machine: a review. Int. J. Comput. Sci. Inf. Secur. (IJCSIS) **6**(3), pp. 181–205 (2009). ACM: ACM Policy and Procedures on Plagiarism, http://www.acm.org/publications/policies/plagiarism_policy
8. Yadav, S.K., Mukhedkar, M.M.: Review on speech recognition. Int. J. Sci. Eng. **1**(2), 61–70 (2013)
9. Luengo, I., Navas, E.: Feature analysis and evaluation for automatic emotion identification in speech. IEEE Trans. Multimedia **12**(6), 267–270 (2010)
10. Wiqas, G., Singh, N.: Literature review on automatic speech recognition. Int. J. Comput. Appl. **41**(8) (2012) (0975 – 8887)
11. Dave, N.: Feature extraction methods LPC, PLP and MFCC in speech recognition. Int. J. Adv. Res. Eng. Technol. **1**(VI) (2013)
12. Hinton, G., Deng, L., Yu, D., Dahl, G., Mohamed, A., Jaitly, N., Senior, A., Vanhoucke, V., Nguyen, P., Sainath, T., Kingsbury, B.: Deep neural networks for acoustic modeling in speech recognition. IEEE Sig. Process. Mag. **29**(6), 82–97 (2012)
13. Prabhakar, O.P., Sahu, K.N.: A survey on: voice command recognition technique. Int. J. Adv. Res. Comput. Sci. Softw. Eng. **3**(5) (2013)
14. Muda, L.: Voice recognition algorithms using mel frequency cepstral coefficient (MFCC) and dynamic time warping (DTW) techniques. J. Comput. **2**(3) (2010)
15. George, K.K., Arunraj, K., Sreekumar, K.T., Kumar, C.S., Ramachandran, K.I.: Towards improving the performance of text/language independent speaker recognition systems. In: International Conference on Power, Signals, Controls and Computation (EPSCICON), 8–10 January 2014
16. Hao, T., Chao-Hong, M., Lin-Shan, L.: An initial attempt for phoneme recognition using structured support vector machine (SVM). In: IEEE International Conference on Acoustics Speech and Signal Processing (ICASSP), 2010, pp. 4926–4929 (2010)
17. Gemmeke, J.F., Virtanen, T., Hurmalainen, A.: Exemplar-based sparse representations for noise robust automatic speech recognition. IEEE Trans. Audio Speech Lang. Process. **19**(7), 2067–2080 (2011)
18. Cheng, X., Duan, Q.: Speech emotion recognition using Gaussian mixture model. In: 2nd International Conference on Computer Application and System Modeling, pp. 1222–1225 (2012)
19. Nidhyananthan, S.S., Kumari, R.S.S.: Text independent voice based students attendance system under noisy environment using RASTA-MFCC feature. In: International Conference on Communication and Network Technologies (ICCNT) (2014)
20. Tan, T.S., Ariff, A.K., Ting, C.M., Salleh, S.H.: Application of Malay speech technology in Malay speech therapy assistance tools. In: Proceedings of IEEE Conference on Intelligent and Advanced Systems, pp. 330–334 (2007)
21. Venkateswarlu, R.L.K., Kumari, R.V.: Novel approach for speech recognition by using self organised maps. In: 2011 International Conference on Emerging Trends in Networks and Computer Communications (ETNCC), Udaipur, pp. 215–222 (2011)
22. Cutajar, M., Gatt, E., Grech, I., Casha, O., Micallef, J.: Comparative study of automatic speech recognition techniques. IET Sig. Process. **7**(1), 25–46 (2013)
23. Ravikumar, K.M., Rajagopal, R., Nagaraj, H.C.: An approach for objective assessment of stuttered speech using MFCC features. ICGST Int. J. Digit. Sig. Process. **9**, 19–24 (2009)

24. Seehapoch, T., Wongthanavasu, S.: Speech emotion recognition using support vector machines. In: 5th IEEE International Conference on Knowledge and Smart Technology (KST), pp. 86–91, Jan 2013
25. Abu Shariah, M.A.M., Ainon, R.N., Zainuddin, R., Khalifa, O.O.: Human computer interaction using isolated-words speech recognition technology. In: International Conference on Intelligent and Advanced Systems, ICIAS 2007, pp. 1173–1178 (2007)

Challenges and Issues in Adopting Speech Recognition

Priyanka Sahu, Mohit Dua and Ankit Kumar

Abstract The area of automatic speech recognition is being discussed from past few decades, and significant advancement is being observed periodically on the automatic speech recognition (ASR) and language spoken systems. However, there are many technological hurdles yet to reach flexible solutions that satisfy the user. This is because of many factors such as environmental noise, paucity of robustness to speech variations (foreign accents, sociolinguistics, gender, and speaking rate), spontaneous, or freestyle speech. To realize the ubiquitous adoption of speech technology, there is need to bridge the space between what speech recognition technologies can convey and what human need from it. To make it up, technology must deliver robust and high-recognition accuracy near to man-like performance so it demands to focus on the challenges in speech technology.

Keywords Speech recognition · Speech modeling · Feature extraction techniques
Speech variations

1 Introduction

Speech is the way of exchanging information and views among human beings. The use of speech as a man–machine interface studied during past few decades, and magnificent progress has been made in the era of speech technology, but there are still many obstacles must be clear to realize the ubiquitous adoption of speech technology. Speech recognition can be stated as a technique of translation of speech signal into text form by using some algorithmic rule implemented as a machine

P. Sahu (✉) · M. Dua · A. Kumar
National Institute of Technology Kurukshetra, Haryana, India
e-mail: er.priyankasahu40@gmail.com

M. Dua
e-mail: mohitdua@gmail.com

A. Kumar
e-mail: Ankitvet@gmail.com

S. S. Agrawal et al. (eds.), *Speech and Language Processing for Human-Machine Communications*, Advances in Intelligent Systems and Computing 664,
https://doi.org/10.1007/978-981-10-6626-9_23

program. There are a lot of commercial products existed over from last twenty years, initially for isolated or digit identification and later for connected words, continuous speech and now active research are going on spontaneous speech. Almost all existing systems are using statistical modeling, including both acoustic and linguistic levels. Basically, ASR categorized by two acoustic models such as (1) word model and (2) phone model. When vocabulary size is concise, we use word model where words are modeled as whole. In case of phone model, despite modeling the complete word, we model only phones.

2 Developments Made in Speech Recognition

The work in the era of speech recognition has been started from recognition of simple phonemes and goes toward the recognition of fluently spoken languages. Table 1 contains some significant efforts that have been done in last few decades [1].

Table 1 Some historical efforts in speech recognition

History: year wise	Contributor	Contribution	Impact
1920–1960s	In 1920	Radio rex machine developed to recognize speech	First machine to recognize speech
	In 1952, Davis at Bell Labs	Developed an automatic speech recognition (ASR) machine for isolated digit recognition	For single speaker
	In 1959, at University College in England	Phone recognizer is developed to recognize four vowels and nine consonants	Spectrum analyzer and pattern matcher are used to make recognition decision
	In 1959, at MIT Lincoln Laboratory	Vowel recognizer is built	Works in speaker independent manner
1960–1970	In 1960s, Suzuki and Nakata at Radio Research Laboratory	Built a hardware vowel recognizer	–
	In 1962, Sakai and Doshita of Kyoto university	Built a hardware phoneme recognizer	–
	In 1963, Nagata and coworkers at NEC Laboratories	Built a digit recognizer hardware	Most notable initial attempt at speech recognition at NEC
	Martin at RCA Laboratory	Develops realistic solutions to problems associated with non-uniformity of timescales in speech events	Reduces the variability of recognition scores
	Vintsyuk in soviet union	Proposed the use of dynamic time wrapping(DTW)	Includes algorithms for connected word recognition

(continued)

Table 1 (continued)

History: year wise	Contributor	Contribution	Impact
1970–1980	In 1973, CMU's Harpy System	Able to recognize speech using a vocabulary of 1.011 words with reasonable accuracy	first to take advantage of finite state machine (FSN), efficiently determine the closest matching string
1980–1990	In 1980, Mosey J. Lasry	Developed a feature-based speech recognition system	Goal to recognize fluently spoken string of words (e.g., digits), problem of connected word recognition is focused
	–	Template-based approach changed to statistical modeling methods (HMM)	
1990–2000s	In 1990s	discriminative training, e.g., minimum classification error (MCE), maximum mutual information (MMI), wavelets, ANN, SVM [6]	Baye's concept-based problems transformed into an optimization problem involving minimization of error; variable time-frequency tiling more closely matches human perception, excellent static nonlinear classifier.

2.1 Comparison Between Various Developed ASR Systems

Various classification techniques and feature extraction techniques have been developed in order to recognize speech, which gives different accuracy while using on different vocabulary size. Here Table 2 is shown for various developed ASR systems for different languages [2].

3 Challenges in ASR Design

Speech technology is going rapidly fit for use but still, it has not been broadly accepted in our living. There are still many technological challenges that must be uncover to realize the full potential of automatic speech recognition technology in multimodal and intuitive man–machine communication. Various issues that affect the accuracy of speech recognition are described in Table 3 [3].

3.1 Some More Challenges to Minimize the Gap Between Man–Machine Speech Recognition [4, 5]

There have been many technological hurdles that got solved but still many more are left that not resolved yet. Some of these hurdles are:

- Minimize the error rate of speech recognizers

Table 2 Comparison between various developed ASR systems

Year	Recognition type	SI/SD/SA	Language	Feature extraction scheme	Classification technique	Performance (%)
2015	Continuous speech	SA	HINDI	MFCC LPCC	HMM	80 78.1
2014	Isolated word	SI	Telugu [7]	MFCC	HMM	96
2012	Isolated word	SI	Punjabi [8]	MFCC	HMM DTW	94–95 94.08
2012	Connected word	SI	Hindi [9]	MFCC	HMM	95.6
2011	Continuous phonemes	SI	TIMIT Corpus-39 classes	MFCC	HMM-MLP	77.83
2011	Continuous phoneme	SI	TIMIT Corpus-39 classes	PLP	SMLP	78.9
2011	Isolated word	SI	Indian	LPCC MFCC	Modified- SOM	88.05 89.27
2011	Isolated word	SI	6-English words	LPCC	RBF MLP	98.69 96
2010	Context—independent phoneme	SI	TIMIT Corpus-39 classes	MFCC	CDHMM	63.07
2009	recognition Continuous word	SI	10-English words	Cepstrum analysis	HMM-RBF	80
2009	Isolated word	SI	Malayalam	DWT WPT	MLP	89 61
2009	Isolated word	SI	50 English words	Subband MFCC	CDHMM-FNN	89.50
2009	Isolated spoken digits	SI	Persian	MFCC and DWT	MLP	98
2005	Isolated word	SI	DARPA RM1	MFCC	HMM-SVM	94.10
2003	Isolated spoken digits	SI	Corpus English SD2	MFCC-WPT	HMM	38.77 56.90
2002	Isolated word	SD	Corpus Urdu	MFCC	MLP	94
2000	Isolated word	SD	Hindi	LPCC	HMM and VQ	84
1999	Continuous phoneme	SI	TIMIT Corpus-39	MFCC	SVM	77.60

SD: Speaker Dependent, SI: Speaker Independent, SA: Speaker Adaptive

Table 3 Common issues in automatic speech recognition

Environment	1. Addition of ambient noise (office machinery, human conversations, industrial plant, etc.) and non-acoustic noise (electronic, quantization, etc.) 2. Signal/noise ratio, working conditions
Speaker	1. Speaker dependent/independent 2. Variation in articulation (stress, emotions, physiological state, etc.) 3. Sex, age
Channel	1. Distortion of signal 2. Band amplitude 3. Echo
Speech variability	1. Voice pitch(high, low) 2. Phoneme production (isolated words, continuous speech, spontaneous speech) 3. Rate of speech: (a) lexically-based measures (b) acoustically-based measures 4. Foreign and regional accents 5. Voice tone (shouted, normal, quiet)
Transducing characteristics (microphone/telephone)	1. May lead to spectrum mismatch 2. Causes discrepancy in recognition
Language characteristics	1. Complex grammar 2. Huge degree of inflection in word 3. Phonetically and acoustically prefixes and suffixes

Error rate can be minimized by focusing on two issues:

- Robustness can be achieved by refining the existing microphone ergonomics. It can improve the SNR (signal-to-noise ratio) up to 12 dB, so efficiently reducing the challenge of noise in processing stages.
- If varieties of sensors are inserted in microphone(s) to perceive speech-related signals can deliver important information to recognizer in order to enhance user's experience and to minimize recognition errors.

- Speech recognition should be more flexible in noisy acoustic environment [10]
- Overwhelming delicate nature of contemporary speech recognition system design, minimize the intrinsic error rate using multimodal system design
- Syntactic rules, vocal tract modeling
- User interface design that enhances user experience, application designs that guide user input workspace by multimodal intercommunication
- Minimization of efforts while switching speech technology application from one domain to next or one language to another language
- Overwhelming the ultimate challenge of designing workable SR systems for casual nature, freestyle in speech
- To furnish speech recognizers with the potential to grasp and to precise errors (recognizers must contain semantic and pragmatic knowledge, as it helps in removal of recognition errors)

- Bring out admissible acoustic criterion, nonlinear time normalization
- Uncover compact units in continuous speech (word/phoneme borderline)
- Setup anchor point; examine pronouncement from left to right; begin with emphasize vowel, linguistic rules, absent/extra present (“uh”) speech unit
- Lack of vocabulary and tight language structure; chance to add new speech phonemes (sounds), co-articulation effects
- Inadequate acoustic details, recognition algorithms
- Consequence of nasalization, sensation, sonority, vibrations, deformation because of speaker’s acoustical habitat, deformation due to conveying systems (e.g., transmitter–receiver), unpredictable environmental conditions
- Robust and adaptive fast learning, obstructive speaker(s)
- Real-time processing, cost productiveness (effectiveness)
- Identify speech when some more competing speech is there
- Cost-effective ways to join recent speaker(s) to existing system.

4 Conclusion

Numerous applications have been deployed with the speech recognition technology, there are several practical limitations have been raised that resist the ubiquitous adoption of speech technology. There have been compromises made in automatic speech recognition to have simple and fast processing systems at the cost of less accuracy. There is need to do more research to remove the gap between man and machine. “How to deal with spontaneous and freestyle conversational speech” are two most faultfinding challenges. ASR systems may be improved by improving acoustic modeling, language modeling, decision making. We need to deploy more speech technologies in future (particularly with the use of multimodality). On our belief, ASR can be highly accurate within the conditions of computations available currently.

References

1. Juang, B.H., Rabiner, L.R.: Automatic speech recognition—a brief history of the technology development. Encyclopedia of Language and Linguistics, pp. 1–24 (2005)
2. Cutajar, M., Gatt, E., Grech, I., Casha, O., Micallef, J.: Comparative study of automatic speech recognition techniques. Signal Process. IET **7**(1), 25–46 (2013)
3. Anusuya, M.A., Katti, S.K.: Speech recognition by machine: a review. Int. J. Comput. Sci. Inf. Secur. (IJCSIS) **6**(3), (2009)
4. Furui, S.: 50 Years of progress in speech and speaker recognition research. ECTI Trans. Comput. Inf. Technol. **1**(2), (2005)
5. Deng, L., Huang, X.: Challenges in adopting speech recognition. Commun. ACM **47**(1), 69–75 (2004)

6. Benzeghiba, M., De Mori, R., Deroo, O., Dupont, S., Erbes, T., Jouvet, D., Wellekens, C.: Automatic speech recognition and speech variability: a review. Speech Commun. **49**(10), 763–786 (2007)
7. Mankala, S.S.R., Bojja, S.R., Ramaiah, V.S.: Automatic speech processing using HTK for Telugu language. Int. J. Adv. Eng. Technol. **6**(6), 2572–2578 (2014)
8. Dua, M., Aggarwal, R.K., Kadyan, V., Dua, S.: Punjabi automatic speech recognition using HTK. Int. J. Comput. Sci. Issues (IJCSI) **9**(4), 0814–1694 (2012)
9. Kumar, K., Aggarwal, R.K., Jain, A.: A Hindi speech recognition system for connected words using HTK. Int. J. Comput. Syst. Eng. **1**(1), 25–32 (2012)
10. O'Shaughnessy, D.: Acoustic analysis for automatic speech recognition. Proc. IEEE **101**(5), 1038–1053 (2013)