

**DEEP NEURAL NETWORK (DNN) BASED MULTILINGUAL
SPEAKER AGE ESTIMATION**

**A THESIS SUBMITTED TO
GRADUATE SCHOOL OF NATURAL AND APPLIED
SCIENCES**

**OF
KOCAELI UNIVERSITY**

**BY
MOHAMMED MUNTAZ OSMAN**

**IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF DOCTOR OF PHILOSOPHY
IN
ELECTRONICS AND COMMUNICATION ENGINEERING**

KOCAELI 2021

**DEEP NEURAL NETWORK (DNN) BASED MULTILINGUAL
SPEAKER AGE ESTIMATION**

**A THESIS SUBMITTED TO
GRADUATE SCHOOL OF NATURAL AND APPLIED
SCIENCES
OF
KOCAELI UNIVERSITY**

**BY
MOHAMMED MUNTAZ OSMAN**

**IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF DOCTOR OF PHILOSOPHY
IN
ELECTRONICS AND COMMUNICATIONS ENGINEERING**

Assoc. Prof. Osman BÜYÜK
Advisor, İzmir Demokrasi University.
Prof. Dr. Ali TANGEL
Jury Member, Kocaeli University.
Prof. Dr. Kemal GÜLLÜ
Jury Member, İzmir Bakırçay University.
Assoc. Prof. Cemal HANILÇI
Jury Member, Bursa Teknik University.
Assoc. Prof. Aysun TAŞYAPI ÇELEBİ
Jury Member, Kocaeli University.

Thesis Defense Date: 05.11.2021

ACKNOWLEDGEMENT

First and foremost I thank the almighty God, Allah the most merciful and the most gracious. Then, I would like to thank my advisors associate Prof. Osman BÜYÜK and Prof. Dr. Ali TANGEL for all their unreserved support throughout my study. It would be unfair not to mention the contribution of Associate Prof. Cemal HANILÇI and Prof. Dr. Kemal GÜLLÜ who gave me critical support and wise advices every semester we meet. I would also like to recognize and forward my sincere appreciation to all the great teachers I had throughout my amazing academic career specially my Turkish teachers, Fatih KIRAN and Demet KILIÇKAN. I would also like to appreciate the unmatched role played by my parents. My father Muntaz Osman who believed in education all his life but never had an opportunity for himself. He spent all his life encouraging his and neighbors' children to go to school. Although my father is still unable to read and write, I believe he is full of wisdom. And my mother Bezo Yusuf is a dedicated supporter of my whole life. I always count on her. I have no words to fully acknowledge her deeds. May Allah give them long, healthy and joyful life.

My wife and life time partner Zebideru Asrat WEYESSA as well as my children Rahmet and Yusuf Mohammed MUNTAZ are always by my side and supported me in my entire journey tirelessly. My wife paid the highest sacrifices in this long process; hence she deserves my heartfelt gratitude and appreciation. Along with my family I would like to appreciate and give due respect to Menbere Asrat WEYESSA, Tesfaye W/GIYORGIS and Melaku Asrat WEYESSA for their unreserved support to me and my family.

I would also like to say thank you to all the great friends I made in Gogeti primary and middle school, Butajira high school, Jimma university while studying my BSc. Huazhong university of science and technology (HUST) and here in Kocaeli during my doctorate study. Specially, Waheeb Salim Abdulrab TASHAN, Hadee MADADUM and Adramane ASSOUMANA in Kocaeli University deserve my appreciation for being there for me when I most needed a friend. I have much respect to all of my friends for their overwhelming encouragement and collaboration. But it would only be fair to mention my dear friends Hamdu KEDIR and Tewodros EYOB at this occasion for their continued motivation before and during my doctorate study. They were my source of strength for my weaknesses on psychomotor skills I have to admit. I also would like to recognize and appreciate the willingness as well as kindness of my good friend Sofiya Ali MEKONNEN. She took responsibility when I desperately needed someone to sign on my agreement document with Jimma University.

I like to appreciate and give credit to my extended family members in Addis, Shashemene, Yirgalem, Jole and around Butajira area especially to the late Haji Mohammed LEJISO (abiye) and Heriya BARGICHO (Eniye). It was in Yirgalem city back in 1991 where I first started my long academic journey with the help of

these two grandparents. My late Enye used to say “If you hadn't come to us when you were a little kid, you wouldn't have succeeded in education” and that is absolutely true. I fully recognize and approve her thought. Most of the kids in my village couldn't make it through. May Allah give both of them eternal peace (Jannah).

My special appreciation goes to Sherefa Haji NURI and Kebede GELETU whom I call father for their kind support in my high school education together with my parents. I actually cannot thank enough all the great people I happen to know in my life. I sometimes even ask myself why people are so kind to me, who am I to deserve such an incredible support.

Finally I would like to recognize the support offered to me by some institutions. Hence I would like to say thank you to Jimma University, Turkish government scholarship council (YTB) and Ethiopian ministry of science and higher education (MOSHE) for their financial and material support. My sincere appreciation also goes to Kocaeli city administration especially to those working with international students association notably Cuneyt ARI, the late Muhammet YAMAN (May Allah give him Jannah) and Fatih KARAÇOBAN. The city administration offered us free transport facility inside the city which needs to be given due credit. I have met several wonderful people in the city during my tenure as a researcher. One of them is Yunus AKYÜREK who supported me at the final stage of my study. Last but not least, I would like to appreciate English academy in Izmit, particularly the hardworking and sociable colleague Merve AYVACIK. They offered me several opportunities to help their students and overcome my financial constraints at the same time.

This PhD. Dissertation is dedicated to five most important people in my life: to my father Muntaz OSMAN who used to carry me all the way to Gogeti primary school located approximately 45 minutes far away from home at times, to my mother Bezo YUSUF who invested her whole life to the wellbeing and success of me and my four siblings, to my equivalent grandparents the late Haji Mohammed LEJISO and Heriya BARGICHO (may Allah give them Jannah) who put hope in my mind and believed in the power and benefit of education and a special dedication goes to my late grandmother Ramete Sheikh ABDO (as we call her DAKO at home and the community calls her ANATO meaning aunty in GURAGE and SILTE communities). May her soul rest in eternal peace. And finally to my beloved country Ethiopia, which has been at a critical juncture recently. I believe we will eventually prevail as a nation and we are destined to greatness.

November 2021

Mohammed Muntaz OSMAN

TABLE OF CONTENTS

ACKNOWLEDGEMENT	i
TABLE OF CONTENTS	iii
LIST OF FIGURES	v
LIST OF TABLES	vii
LIST OF SYMBOLS AND ABBREVIATIONS	ix
ÖZET.....	xi
ABSTRACT.....	xii
INTRODUCTION	1
1. REVIEW OF SPEECH PROCESSING APPLICATIONS AND RESEARCH DYNAMICS	11
1.1. Speech for Age Recognition.....	11
1.2. Speech Generation and Perception	13
1.3. Procedural and Rule-based Classical Programming versus Artificial Intelligence	20
1.4. Literature Review and the Research Dynamics.....	22
2. ADAPTED AND PROPOSED FEATURE EXTRACTION TECHNIQUES	32
2.1. Introduction	32
2.2. Time and Frequency Domain Analysis of Speech Signal	33
2.2.1. Pre-emphasis	33
2.2.2. Windowing.....	34
2.2.3. Time domain analysis	37
2.2.4. Frequency domain analysis.....	39
2.3. Filter Bank Based Features.....	41
2.3.1. Mel-frequency cepstral coefficient (MFCC)	41
2.3.2. Rectangular filter cepstral coefficient (RFCC).....	45
2.3.3. Linear frequency cepstral coefficient (LFCC).....	46
2.3.4. Inverted mel-frequency cepstral coefficient (IMFCC)	47
2.3.5. Parabolic filter mel-frequency cepstral coefficient (PFMFCC).....	47
2.4. Phase and Sub-channel Based Features.....	50
2.4.1. Sub-band spectral flux coefficients (SSFC).....	50
2.4.2. Sub-band centroid magnitude and frequency (SCMF).....	51
2.4.3. Relative spectral transform perceptual linear prediction (RASTA-PLP)	51
2.4.4. Cosine phase	52
2.4.5. Modified group delay (MODGD).....	52
2.5. Feature selection, Feature Fusion and Dimensionality Reduction	53
2.5.1. Union selection of feature sets.....	54
2.5.2. Principal component analysis (PCA).....	57
2.5.3. Linear discriminant analysis (LDA)	58
2.5.4. Feature fusion	61

3.	EMBEDDING WITH MACHINE LEARNING AND DEEP LEARNING MODELS	62
3.1.	i-Vector Embedding	62
3.2.	Deep Learning-Based Embedding.....	63
3.2.1.	Introduction to deep learning.....	63
3.2.2.	x-Vector embedding	65
4.	CLASSIFICATION AND REGRESSION MODELS	69
4.1.	Gaussian Mixture Model (GMM)	69
4.2.	Cosine Distance Scoring with i-Vector (CDS).....	71
4.3.	Probabilistic Linear Discriminant Analysis with i-Vector (PLDA).....	72
4.4.	Deep Neural Network (DNN) Based Classifiers.....	75
4.4.1.	x-Vector deep neural network architecture for classification	79
4.4.2.	Long short-term memory (LSTM) networks for classification	80
4.5.	Regression Models	81
4.5.1.	Linear regression.....	81
4.5.2.	Non-linear and least square support vector regression (LSSVR).....	87
4.5.3.	LSTM for speaker age regression.....	93
5.	EXPERIMENTAL SETUP	95
5.1.	Databases	95
5.2.	Classification and Regression Experimental Setups	97
6.	RESULTS AND DISCUSSION.....	101
6.1.	Results	101
6.1.1.	Performance evaluation of CDS, GMM and PLDA classifiers on matched-language baseline scenarios.....	101
6.1.2.	Performance evaluation of CDS, GMM and PLDA classifiers on bilingual, multilingual and cross-language scenarios	106
6.1.3.	Regression results and speech duration analysis	111
6.1.4.	Performance evaluation of deep learning based classifiers	115
6.2.	Discussion.....	117
6.2.1.	PFMFCC versus MFCC for speaker age classification.....	118
6.2.2.	Unexpected effect of VAD	119
6.2.3.	Performance of feature fusion.....	120
6.2.4.	Limitations, solutions and findings.....	121
7.	CONCLUSION	125
	REFERENCES.....	128
	PERSONAL PUBLICATIONS AND ACHIEVEMENTS	138
	RESUME	139

LIST OF FIGURES

Figure 1.1.	The vocal tract and the process of speech production.....	16
Figure 1.2.	The acoustic tube model and the vocal tract area function	16
Figure 1.3.	Speech production model.....	18
Figure 1.4.	Speech perception process	19
Figure 1.5.	Inception, generation, propagation and interpretation of speech.....	19
Figure 1.6.	Classical programming versus machine learning paradigm.....	21
Figure 2.1.	General block diagram of pre and post signal processing operations	32
Figure 2.2.	Framing and feature extraction using windows	35
Figure 2.3.	Framing windows.....	36
Figure 2.4.	Framing with 50% overlap.....	37
Figure 2.5.	Autocorrelation function	39
Figure 2.6.	Triangular filter banks for MFCC	49
Figure 2.7.	Parabolic filter banks for PFMFCC	49
Figure 2.8.	Dimensionality reduction methods	54
Figure 2.9.	Classification of Iris data with two features using nearest mean	56
Figure 3.1.	i-Vector extraction	62
Figure 3.2.	Multilayer deep neural network structure	64
Figure 3.3.	Time delay neural net (TDNN) Computation with sub- sampling (red) and without sub-sampling (blue+red).....	66
Figure 3.4.	TDNNs to softmax end-to-end speaker age estimation	68
Figure 4.1.	Gaussian distributions and mixture.....	70
Figure 4.2.	Projection of observed features in to latent space.....	73
Figure 4.3.	Overall process diagram of CDS and PLDA Classifiers	74
Figure 4.4.	Biological neural network.....	75
Figure 4.5.	Deep neural network sample with two layers	77
Figure 4.6.	General block diagram for x-vector architecture embedding	79
Figure 4.7.	Classification and regression with LSTM.....	80
Figure 4.8.	Linear regression (Picture credit to Wikipedia).....	82
Figure 4.9.	Gaussian mixtures and kernel functions	89
Figure 4.10.	Least square support vector regression (LSSVR)	91
Figure 4.11.	Peephole connections in LSTM cells.....	93
Figure 5.1.	General block diagram that shows the overview of experiments in this study.....	98
Figure 5.2.	Speech length in terms of number of frames for age estimation	98
Figure 6.1.	Graphic representation of evaluation results for female test set in a) cosine score b) GMM and c) PLDA classifiers with aGender database	104

Figure 6.2.	Graphic representation of evaluation results for male test set in a) cosine score b) GMM and c) PLDA classifiers with aGender database	105
Figure 6.3.	MAE of LSSVR expressed along increasing number of frames for male aGender dataset.....	112
Figure 6.4.	ρ as frames increase for LSSVR for male aGender dataset	112
Figure 6.5.	MAE of LSSVR expressed along increasing number of frames for female aGender dataset.....	113
Figure 6.6.	ρ as frames increase for LSSVR over female aGender dataset.....	113
Figure 6.7.	Effect of VAD on the PLDA classifier for male and female datasets of aGender database from simulation results	119
Figure 6.8	Performance evaluation results of feature fusion of seven feature sets on three classifiers on seven and three age class arrangements	121
Figure 6.9.	Performance comparison of matched-language, multilingual, bilingual and cross-language training scenarios for speaker age classification	124

LIST OF TABLES

Table 2.1.	List of filter banks in magnitude spectral feature extractions.....	50
Table 3.1.	Context specification of the TDNN shown in fig. 4 above.....	67
Table 5.1.	Distribution of speakers along development, training and test sets in each class of the aGender database.	95
Table 5.2.	Distribution of utterances along development, training and test sets in each class of the aGender database.	96
Table 5.3.	Distribution of utterances along development, training and test sets in each class of the Age-Vox-Celeb database.	97
Table 5.4.	Distribution of speakers along development, training and test sets in each class of the Turkish database.....	97
Table 6.1.	Comparing the proposed PFMFCC in female datasets of the aGender database with and without VAD.....	102
Table 6.2.	Comparing the proposed PFMFCC in male datasets of the aGender database with and without VAD.....	103
Table 6.3.	Comparing the proposed PFMFCC for female and male datasets in the Turkish database.....	106
Table 6.4.	Bilingual training tested with German and Turkish female and male datasets for speaker age classification.....	107
Table 6.5.	Multi-language training performance evaluation for female and male datasets with German (aGender), Turkish and English (Age-Vox-Celeb) databases.....	108
Table 6.6.	Cross-language and matched language performance evaluation for female and male datasets trained with German, Turkish and English databases tested with German, Turkish and English test sets.....	109
Table 6.7.	Performance comparison of best matched-language classification accuracies with multilingual and cross-language scenarios.....	110
Table 6.8.	Performance evaluation in a) MAE b) ρ of feature+i-vector+LSSVR method for male dataset.....	111
Table 6.9.	Performance evaluation in a) MAE b) ρ of feature+i-vector+LSSVR method for female dataset.....	111
Table 6.10.	i-Vector followed by LSSVR performance evaluation for utterance length mismatch in terms of MAE for female dataset. (Rows are training and columns are test frames).....	114
Table 6.11.	i-Vector followed by LSSVR performance evaluation for utterance length mismatch in terms of MAE for male dataset. (Rows are training and columns are test frames).....	114
Table 6.12.	Performance of LSSVR model on short, medium and long utterances for female and male datasets.....	115
Table 6.13.	Cross-gender speaker age evaluation using x-vector neural network architecture.....	116

Table 6.14.	Cross-language and cross-gender speaker age evaluation using x-vector neural network architecture.....	116
Table 6.15.	Performance evaluation of 5 classifiers with MFCC sequences for speaker age classification on a) female b) male datasets respectively	117
Table 6.16.	MFCC versus PFMFCC	118



LIST OF SYMBOLS AND ABBREVIATIONS

f	: frequency
$mel(f)$: Mel Scale of frequency f
$mel^{-1}(m)$: Inverse of Mel Scale
$lan()$: Natural Logarithm with base e
$exp()$: Natural exponential Function with $e = 2.7183$
i	: Index which stands for 1 st , 2 nd , ..., 10 th Filter bank Function
k	: Frequency bin used in FFT
$V_i[k]$: Filter bank function at index i and frequency bin k
A_i	: Normalization Factor at the i^{th} filter
$MF[i]$: Mel-frequency Spectrum at analysis time n
$X(n, k)$: Periodogram (spectrogram) estimate of the power spectrum at analysis time n
f_s	: Sampling frequency
N	: Number of FFT points (bins)
$MFCC[m]$: Mel-frequency Cepstral Coefficient at index m
ϵ	: Residual noise
Σ	: Covariance
m	: Global mean
x_i	: R-dimensional Observation ($R \times 1$)
T	: Total Variability matrix with dimension $R \times N$
ω_i	: Latent factor or i -vector at index I of dimension N which is ($N \times 1$)
ρ	: Pearson's correlation coefficient

List of Abbreviations

API	: Application Program Interface
CDS	: Cosine distance scoring
CNTK	: Microsoft Cognitive Toolkit
DCT	: Discrete Cosine Transform
DFT	: Discrete Fourier transform
DNN	: Deep Neural Network
DSP	: Digital signal processing or digital signal processor
DTFT	: Discrete time Fourier transform
FFT	: Fast Fourier Transform
GMM	: Gaussian Mixture Model
GRNN	: General regression neural network
HMM	: Hidden Markov Model
IMFCC	: Inverted Mel-frequency Cepstral Coefficient
kHz	: Kilohertz

KNN	: K-Nearest Neighbor
LDA	: Linear Discriminant Analysis
LFCC	: Linear Frequency Cepstral Coefficient
LPC	: Linear Predictive Coding
LSSVR	: Least Square Support Vector Regression
MAE	: Mean absolute error
MFCC	: Mel-frequency Cepstral Coefficient
MODGD	: Modified Group Delay
Ms	: Millisecond
NIST	: National Institute of Science and Technology
Ph.D.	: Doctor of Philosophy
PLDA	: Probabilistic LDA
PLP	: Perceptual Linear Predictive
RASTA	: Relative Spectral Transform
RASTA-PLP	: Relative Spectral Transform Perceptual Linear Prediction
RFCC	: Rectangular Filter Cepstral Coefficient
SCMF	: Sub-band Centroid Magnitude and Frequency
SSFC	: Sub-band Spectral Flux Coefficients
STFT	: Short Time Fourier Transform
SVM	: Support Vector Machine
TDNN	: Time delay neural network
VAD	: Voice activity detection
WCCN	: Within-Class Covariance Normalization
WSNMF	: Weighted Supervised Non-Negative Matrix Factorization

DERİN SINIR AĞI (DSA) TABANLI ÇOK DİLLİ KONUŞMACI YAŞ TAHMİNİ

ÖZET

Finans, perakende ve diğer sektörler için çevrimiçi faaliyetlerin çarpıcı bir şekilde büyümesiyle birlikte, internet kullanıcılarının uzaktan profillenmesi çok önemli bir gereklilik haline geldi. Konuşmacı yaşı tahmini, özellikle uzak kullanıcılar için bu ihtiyacın etkin bir şekilde ele alınmasına büyük ölçüde yardımcı olabilir. Konuşmacı yaş tahmini, konuşmayı kullanarak yaş sınıflarını ve ya gerçek yaş değerlerini tahmin etmek olarak tanımlanabilir. En önemlisi, çocuklar internetteki grafik ve şiddet barındıran içeriklere genellikle fark edilmeden eriştikleri için, çocukların korunmasında konuşmacı yaşı tahmin sistemleri kullanılabilir.

Bu çalışmada, farklı sınıflandırma ve öznitelik çıkarma teknikleri konuşmadan yaş sınıflandırma ve regresyon problemleri için kullanılmıştır. Bu özniteliklerin çoğu, konuşmacı yaşı tahmini için daha önce kullanılmamıştır.

Parabolik filtre mel frekansı kepsral katsayısı (PFMFKK), mel frekansı kepsral katsayılarında (MFKK) filtre bankalarının (bant geçiren filtre dizisinin) şeklini değiştirerek yeni bir öznitelik çıkarma yöntemi olarak önerilmiştir. PFMFKK, uyarlanmış tüm öznitelik setlerine kıyasla kadın ve erkek veritabanları için olasılıksal doğrusal ayırım analizi (ODAA, PLDA) sınıflandırıcısı ile en iyi performansı sunmuştur. Ayrıca diğer sınıflandırıcılarla da karşılaştırılabilir sonuçlar vermiştir. Konuşmacı tanıma için önerilen i-vektör ve x-vektör vektör gösterimleri de yaş tanıma problemine uygulanmıştır.

Bu tezde ayrıca veri tabanları arasındaki dil ve ortam farklılığının yaş tanıma performansı üzerindeki etkisi incelenmiştir. Bu amaçla Türkçe, Almanca ve İngilizce üç farklı veri tabanı kullanılmıştır. Bu veri tabanlarının hedef dilleri ile birlikte toplandıkları ortamlar/geri plan gürültü oranları da birbirinden oldukça farklıdır. Deneysel sonuçlar, çok dilli eğitim senaryosunun, tek dilli senaryoya göre yaş tahmini performansını çok fazla etkilemediğini, ancak diller arası eğitim/test senaryosuna kıyasla performansı önemli ölçüde iyileştirdiğini göstermiştir.

Anahtar Kelimeler: Çok Dilli Eğitim, Derin Öğrenme, Konuşmacı Yaş Tahmini Öznitelik Füzyonu, Parabolik Filtre Bankası.

DEEP NEURAL NETWORK (DNN) BASED MULTILINGUAL SPEAKER AGE ESTIMATION

ABSTRACT

With the dramatic growth of online activities for finance, retail and other sectors remote profiling of internet users has become a crucial necessity. Speaker age estimation can greatly help in effectively addressing this need especially for remote users. Speaker age estimation can be defined as predicting either age classes or actual age values exploiting speech. Most importantly, speaker age prediction systems can be applied in safeguarding children as they usually access graphic and violent contents on the internet unnoticed.

In this study, several feature extraction techniques are adapted and employed on selected classification and regression models. Most of these features have never been used for speaker age estimation. These features are used as input to selected machine learning and deep neural network (DNN) models over age labeled multilingual databases. i-Vector and x-vector embedding are applied for fixed dimensional representation.

Parabolic filter mel-frequency cepstral coefficient (PFMFCC) is proposed as a new feature extraction method by modifying the shape of the filter banks in mel-frequency cepstral coefficients (MFCC). PFMFCC offered the best performances with probabilistic linear discriminant analysis (PLDA) classifier for female and male databases compared to all adapted feature sets. It also showed comparable results with other classifiers.

Multilingual settings are established to introduce diversity in language and are observed making differences especially when there is language mismatch. Experimental results indicate that multilingual training setup does not affect the performance of speaker age estimation in single language approaches much, but it improves the performance compared to cross-language evaluations significantly.

Keywords: Multilingual Training, Deep Learning, Speaker Age Estimation, Feature Fusion, Parabolic Filter Bank.

INTRODUCTION

The ultimate aim of research is to find out ways, methods and solutions to specific problems that can improve the lives of a human society. Hence, research is a long journey of finding answers to a series of “why” and “how” questions. Discovering the causes of problems is half of the solution. The common saying ‘the devil is in the details’ reflects the challenges during the research process. Research brings new perspectives in to light to solve specific problems in many disciplines. In this sense it is a never ending journey of enquiring answers to the very fundamental questions of “why” and “how”. The “why” questions are usually associated with analysis of problems whereas, the “how” wing often focus on synthesis of solutions.

Characteristics and Research Dynamics of Speech Signal

Speech is made up of both universal and language or culture-specific aspects. The universal aspects are inherently paralinguistic in nature [1]. The balance between these properties is still open to debate. As a matter of practical reality, it is beneficial to improve our understanding of this balance. It can be used to develop multi-lingual speech processing systems and utilize cross-language sharing. This eventually helps to increase the number of languages available for certain speech technologies. In addition, it helps to make the technologies versatile with respect to languages.

Multilingualism, which refers to the use of more than one language by an individual speaker or by a group of speakers, represents an area of significant opportunities for automatic speech-processing systems. Although multilingual societies are commonplace and could be by far the majority worldwide compared to monolingual individuals, the majority of speech processing technologies are developed with a single language in mind mostly English [2]. In Asia and Africa alone, there are more than 4000 languages and the chance of people speaking more than two of these languages, is highly likely.

As a step towards improved understanding of multilingual speech processing, the current contribution investigates on how para-linguistic aspect of speech depends on the language spoken [3]. Para-linguistic aspect of speech include tone, pitch of voice and speaker age. The gap in language diversity requires additional research to make speech processing applications scalable in terms of languages.

The question, “How language emerged in to human evolution?”, may never have a complete answer according to Dr. C. George Boeree, an American psychologist and professor emeritus at Shippensburg University who specialized in personality theory and the history of psychology [4]. It is one of the most difficult questions to give a satisfactory answer. However some scholars believe that language emerged in human society as a result of some kind of social transformation by generating unprecedented levels of public trust. In addition there are several theories which argue on the origin of language [5].

It is a no-brainer that speech differs across cultures and languages worldwide in a multiple of ways, ranging from acoustic phonetics through grammar, vocabulary and metaphor to pragmatics and discourse strategies. The differences involving metaphor or acoustic phonetics may be pronounced even for cultural groups that share the same language, whereas other factors such as grammar tend to encompass a larger set of speakers. However, little attention is given to the effects of culture on speech processing tasks comparatively. English, German and Turkish languages are selected in this study based on availability of speech data to investigate speaker age estimation across different languages as well as multilingual approaches to mitigate language mismatches during evaluation.

Various studies have proposed several methods on speaker age estimation. What we can understand from most of these literatures is that age estimation from speech is very challenging due to its stochastic nature. Speaker age prediction is even more difficult for people. Some of the mechanisms that people can use to predict age include: looking at faces, listening to speech, examining maturity level and others. Apart from such subjective estimations; scientists have made efforts to estimate age from a DNA test [6]. They argued that if they could measure the length of a person's telomere, they would be able to tell his/her age [7]. After all, the more times a cell

divides, the shorter its DNA will be. And the older the person will be, the more times the person's cells will have divided. Although they couldn't determine the person's age from DNA test, it really helped scientists to determine the ethnicity, family relationships and gender. A team of researchers estimated age using 200 nano grams of DNA for each age prediction. The team found its margin of error was 3.75 years for blood samples and 4.86 for teeth. Roughly 80% of the estimations were within five years, either older or younger [6].

Speaker age estimation is the extraction of age information from speaker's utterance. Feature extraction and selecting effective features that represent the speaker's age characteristics uniquely are keys in speaker age classification and regression. Another equally essential stage is the design of a suitable classification or regression method. Classifiers use the features generated through a series of operations to predict the speakers' age. These operations are applied on audio signals. The focus of this research is on finding distinctive feature sets that are able to represent utterances such that selected classifiers can recognize the age group of the speaker with better accuracies than previous studies. In addition, the design of a suitable classifier or regression model plays a major role in predicting the speaker's age. This research investigates different classifiers and regression techniques to enhance age only classification and prediction for each gender as well as age plus gender classification for seven class scenario.

Much of this study gives an in-depth focus for feature extraction and tries to examine performance of some classification and regression models in the broad artificial intelligence for speaker age estimation on single as well as multi-language databases. However, classification and regression schemes have been dealt in a great number of studies in the past which only need adaptation rather than invention in our work. In addition to investigating the performance of certain feature sets, a new feature set called parabolic filter mel-frequency cepstral coefficient (PFMFCC) is proposed in this study. The majority of the adapted features have never been employed for speaker age estimation to the best of our knowledge. Choice of classifiers or regression techniques plays a vital role regardless of which feature set is applied to them. Keeping this in mind we treated some classical and deep neural network (DNN) models in this work.

In a wide spectrum of studies speech features are mainly categorized in to spectral, prosodic and glottal. Spectral features are those generated as a result of spectral analysis of speech. Spectrum refers to the distribution of energy as a function of frequency for a particular sound source. Prosodic features express the rhythm and intonation of a language. The term prosodic refers to the way a speaker's voice rises and falls.

The motivation for age recognition from a speaker's utterance comes from the fact that the vocal tract anatomy changes considerably in the life time of the person. DNN algorithms are proposed recently, that can generate important features to associate speech with age. A well-designed classifier or regression model is equally demanded for this task. Speaker age classification is a crucial issue for targeted advertising in the 21st century as online activities in finance, retail and other sectors have become certainly important.

One way to recognize a person's age is through speech. Speech is one of the ways which enable us to estimate age of a person in addition to appearance. Age recognition together with gender, accent and emotional recognitions has got a wide range of applications in language learning, remote advertising (tele-marketing), criminal investigations, automated health, education and human-computer interaction(HCI) [8] .For all these application areas, systems can be customized based on speaker age category. This will highly improve user satisfaction level. Games can be designed based on age group, commercials can be broadcasted for specific age categories, and medical diagnosis can be carried out according to speaker's age [9].

Speaker age estimation can help speaker recognition or verification efforts in contemplating speaker's speech over the years lived. This is extremely helpful especially in identifying criminals who have stayed behind public attention for long years. Criminals change their appearances and speech patterns. Although face changes are quite difficult to trace as it can also be engineered, speech changes can be treated using collaborative effort of speaker age estimation and speaker recognition techniques.

Background noise, accent variation, speech duration, text-dependent or text-independent control variable, recording device variation, channel and space variability, and other related factors make speaker age classification as one of the most challenging tasks in speech processing research. Speaker age classification consists of feature extraction and classification. A carefully designed feature extraction technique is not only able to extract age related features from speech but also combat the effect of background noise as the noise coming from the surrounding is unavoidable [10]. Classification in this context, is grouping training samples in discrete categories and to develop models for each category.

Generally popular features such as mel-frequency cepstral coefficient (MFCC) [11], energy, relative spectral transform (RASTA) [2], speech rate [12], RASTA-perceptual linear prediction (RASTA-PLP)[13], are used in age classification. In addition to these features, other features can also be calculated using prosodic or glottal characteristics of speech utterances. Four variants of MFCC, two sub-channel based features, two phase-based spectral features and RASTA-PLP are employed in this study.

The i-vector is first proposed for speaker verification and they are successfully applied to age classification task in recent studies [14]. In the study, i-vectors corresponding to each age class are averaged in the training phase. The cosine distance between each test sample and each target age class i-vector is computed during the test. A similar approach is followed in our work as well. The study carried out in [15] achieved state-of-the-art performance on the aGender database. A feed-forward DNN for age classification using features extracted from utterances is proposed in another study which tries to combine long-term and short-term features [16]. In this method, Gaussian mixture model (GMM) super-vectors are fed into the DNN similar to the GMM/SVM. A DNN age classification method that combines database of German and Turkish speech utterances is proposed in [17]. This method achieved an absolute improvement of 7% over GMM classifier.

Weighted supervised non-negative matrix factorization (WSNMF) is used together with a general regression neural network (GRNN) for age estimation and gender detection from speech [18]. This matrix is trained with GMM weight super-vectors.

GRNN is preferred over other neural networks since it does not demand an iterative training and it is more effective if it is used for sparse data. A performance better than chance level, is obtained using this experiment.

Chronologically, the first major task in this study was to summarize the performance evaluation of three classifiers (GMM, Cosine Distance Scoring (CDS) and PLDA) using 10 feature sets for speaker age classification. Most of these feature sets are used for replay and spoofing attack detection in a previous study[19]. GMM and i-vector classifiers are employed on these feature sets to detect genuine and spoofed utterances. The constant-Q cepstral coefficients (CQCC) features with i-vector classifier was found to offer the smallest equal error rate (EER) which is 21.38% on the evaluation set used in the experiment. The aGender German [20], Turkish [17] and Age-Vox-Celeb English [21] databases are used in our study. The task is performed on male and female genders separately.

Motivation of the Study

Back in 2016 we started to re-examine and explore the capabilities of classical and modern classifiers and function approximation approaches. The aim of this investigation was to apply these methods on speech processing applications. We identified three interesting areas that we can apply deep learning and machine learning algorithms to speech processing problems:

1. Speech based criminal investigation
2. Speech for diagnosis of breathing system related health problems
3. Speaker age estimation and classification

Due to shortage of access to appropriate data we declined not to proceed on the first two topics. With the outbreak of COVID19 pandemic, the second problem could have been a ground breaking research in its outcome due to its level of necessity. However, based on ease of access to suitable database we decided to conduct speaker age estimation and classification. In addition, the explosion of violent contents on the internet demanded our mind to devise a method that can limit access to these contents. These internet contents are inappropriate and very abusive for children and young people. With this in mind, identifying users based on their speech as children, young, adults or elders remotely can save children and the youth from psychological

trauma while watching violent online resources. Moreover, its commercial benefit in targeting users remotely based on their age class is what convinced us to make our choice.

Kids these days can easily access violent and highly graphic websites that can affect their mental development. Placing age limit requirements is extremely demanded in such digital platforms. Most websites require users if they are not a robot prompt which can be successfully completed by even children. But speech input must be demanded to prove a user is not below the required age. Speech is highly secure compared to text and image data regarding user age information.

Speech is more natural and if effectively implemented, it can easily be utilized for remote applications and most importantly it is more convenient and reliable compared to data communication. In commercials an automatic advertisement is very common these days. It would be smarter if the automatic system could predict the age of the person on a phone call. Customers feel satisfied when their preferences have been foreseen and understood. Therefore, speech is the best choice for age information extraction. Considering the current challenges due to the COVID19 pandemic, speech is undeniably preferable despite the fact that age information can be retrieved from various other ways including facial images [22].

There is an old but re-emerging phenomenon called ageism [23]. Just like other sectarian thoughts this could also be a threat to mutual coexistence of human society. Ageism is selectively favoring or disregarding people based on their age [24]. It can be casual or in some societies it might even be systematic. Robert Neil Butler used this term for the first time in 1969 to describe the discrimination against older people. There is a popular expression in Ethiopian society related to this idea which describes older people as “the 1960s generation”. In fact the expression is not only related to ageism, it also refers a political rhetoric. Sectarianism is generally an uncivilized, demonizing and counterproductive to mutual coexistence, peace and happiness of the human society.

This study is conducted with the aim of improving estimation metrics in general, or increasing classification accuracies and reducing regression errors in particular for speaker age estimation using specifically the aGender German and Turkish databases

The Turkish database is collected mainly for voice conversion that includes age information[17]. Therefore, this research work is basically a classification as well as regression problem involving multilingual data for speaker age estimation. Trainings are carried out with single language as well as multi-language datasets. Evaluations are also considered for matched, cross-language and multilingual scenarios.

Freedom of speech versus hate speech

Most people use social media platforms such as Facebook, Twitter, Instagram and others to express their free opinion [25]. Unfortunately some contents expressed as a free thought could instigate violence and others demonize individuals and even societies collectively. These are acts of hate speech [26]. Hate speech has caused the death of millions in Africa and many parts of the world. It is easily accessible to billions of internet users. Children and young people are the most likely to be victims of hate speech as they lack maturity.

Social media platforms need to have plans to balance between freedom of speech and hate speech. People can have the right to express their opinion to the extent of hate [25]. But these platforms must develop mechanisms to have monopoly on who can view selected contents and who must not view them. Age classification based on speech can greatly contribute to this effort. It is not only authentic but also secure to prompt users to utter their speech for a few seconds and make a decision whether to grant or deny access based on speaker age.

Children can greatly benefit from such careful design of social media platforms as it protects them from viewing harmful, hateful and violent contents. Speaker age classification especially, for children recognition, can be employed with less challenge compared to young, adult and old speakers. This is mainly because their speech characteristics is more distinct, separable and contains the highest fundamental frequency [27]. The average fundamental frequency F_0 generally decreases with age across both male and female children of age from 6 years to 16 years old [28].

Scope and Main Contribution of the Study

This study is confined to age estimation in general, classification and regression in particular based on speaker utterances. The study consists of two databases namely; the aGender database which consists of 47 hours of German speeches uttered by speakers of age 7 to 80 years old [20] and a Turkish database mainly collected for voice conversion. Some speech data is added from Age-Vox-Celeb database in order to include English speakers in certain scenarios. The study mainly focuses on front end analysis of speech aimed at finding suitable feature sets for speaker age estimation. Our study began with classification but later extended to estimation including some regression models. However, classification is generally believed to offer more benefits and more feasible with small database than estimation. Although good speaker age estimation leads to an acceptable accuracy level of classification, it requires a relatively larger database than classification. Our experiments are carried out independently on both genders as well as on a mixed database of consisting male and female utterances for training and testing.

In this thesis, several feature extraction schemes are employed for speaker age estimation with selected classification and regression models. The majority of the feature sets have never been used for speaker age estimation before to the best of our knowledge. Except few the majority of adapted features performed comparatively well compared to conventional feature sets. And quite few including phase-based spectral features have surprisingly outperformed the popular MFCC feature with certain classifiers. We carried out the experiments using Matlab, Python [29], and Kaldi toolkit and verified better feature sets for certain classifiers [30].

We proposed a new feature set based on previously implemented feature extraction techniques. We used parabolic shaped filter banks instead of the very common triangular one implemented in MFCC. For ease of nomenclature we named the new feature sets as parabolic filter MFCC (PFMFCC) based on the shape of the filter bank. This new feature has improved the accuracy of speaker age classification with PLDA classifier and offered comparable results in other classifiers and regression models.

We have also applied state of the art methods to represent utterances with fixed dimensional vectors; i-vector and x-vector. We used classical as well as neural network classification and regression models. In addition, this study verified the positive impact of utterance length for speaker age estimation. On top of that we further investigated impact of mismatch in length of utterances within training and test datasets over speaker age estimation performance.

In summary, this research work combines techniques from digital signal processing (DSP) particularly, speech processing and artificial intelligence (AI) to predict speaker age either in terms of age groups or actual chronological age values using short utterances. The AI techniques specifically include selected machine learning and deep learning classification and regression models. Utterances are taken from three databases of English, German and Turkish language speakers.

This Ph.D. thesis is organized as follows: chapter 1 presents, overview of related literatures and developments in speech processing research, chapter 2 discusses the front end analysis techniques and the proposed PFMFCC feature set in detail, chapter 3 briefly examines the two embedding; i-vector and x-vector and chapter 4 presents classification and regression schemes used. Following the methodology sections, chapter 5 presents the experimental setups, procedures employed and parameter specifications, whereas results, discussions and conclusions are presented in chapter 6. The last chapter relates our hypothesis with experimental results and closes the study with concluding remarks at its final sub section eventually.

1. REVIEW OF SPEECH PROCESSING APPLICATIONS AND RESEARCH DYNAMICS

1.1. Speech for Age Recognition

Speech contains paralinguistic information such as speaker age in addition to the usual linguistic contents. When we break it to the level of phonemes, it takes approximately 100 milliseconds for people to utter a single phoneme. A phoneme is the smallest unit of speech. The English and Turkish languages for instance, have 48 and 29 phonemes respectively. The typical number of phonemes in the world's languages ranges from 30 to 50. Therefore, we need utmost 6 bits to represent all the phonemes in a certain language. Even though it could vary across age, people can produce 60 bits of information in a second through their speech. However, the actual information content is notably higher as speech also contains essential information about identity, gender, age, health status, smoking status, alcohol level, accent, the rate of speaking, loudness etc.

Human speech is rich in information. Efforts have made it possible to use some of the potential applications of speech processing. Features extracted from an audio can convey a vast range of information. Typical applications include speech recognition, speech synthesis, speaker recognition, and others. In relation to these applications we have been conducting a series of experiments and simulations in our laboratory to use these applications for different purposes. Among these efforts speaker age recognition and vocal tract related illness detection using features extracted from an audio have captured our interest. While the former application is quite possible since we have organized data and we have also access to a standard database, the later could have taken years of data collection efforts. We pursued the age identification due to free database access whereas; the illness detection project could be our future research focus. .

Popular applications of speech communication include but not limited to:

- Digital transmission and storage
- Speech synthesis
- Speaker recognition, verification or identification
- The popular speech recognition
- Handicap aids
- Signal quality refinement
- Speaker emotion, accent, gender and age recognition
- Speech assisted automations

The main reason why speech is preferred for information extraction in this research is because it is safe, reliable and remotely exploitable. In addition, there are situations where we could be forced to know something from an utterance of an individual. In case of age verification for instance a company which advertises its product through automatic phone calls, it would be more desirable if the system could recognize the approximate age of the intended customer. In fact it is less likely for a person to be shy to tell a system. But customers would feel happy if no system bothers them about their age. In criminal investigation, information extracted from speech of a suspect could lead to verify his identity. Estimating the age group in case of criminal investigation could reduce the scope of suspects. It helps to narrow down the age range of suspects.

The reason why speech based research has remained an active area of study is mainly because of its versatile applications. Its applications cover speech recognition, speaker recognition, speaker age recognition, speaker emotion recognition, speech analysis, speech synthesis, speech enhancement, speech print (voice print) and more. A breakthrough in speech processing research can boost many sectors of our modern life style. So far much has been done in speech recognition and a significant development is also carried out in speaker recognition too. Unlike these two areas much more effort is needed in specific areas such as age, emotion and accent recognition from speech utterances. Natural language processing (NLP) is another major area of research which attracted the attention of a significant number of

scholars. We cannot ignore the efforts that have been delivered in the two decades since the new millennium.

1.2. Speech Generation and Perception

The motivation for identifying the age of a speaker from his/her voice comes from the fact that the vocal tract anatomy changes as the person gets older. There must be a way that can be used to find out important features which associate people of the same age. Not only a feature, but we also need a well-designed classifier for this task. Age classification is a crucial issue in the twenty first century as online activities such; as online shopping, online advertising, electronic commerce, retail, etc. are getting increasing importance. One way to recognize a person's age is through speech. Speech is one of the ways which enable us to estimate age of a person in addition to appearance. Age recognition together with gender, accent and emotional recognitions has got a wide range of applications in language learning, remote advertising (tele-marketing), criminal investigations, automated health, education and human-computer interaction(HCI) [31].For all these application areas, systems can be customized based on speaker age category. This will highly improve user satisfaction level. Games can be designed based on age group central requirement, commercials can be broadcasted for specific age category, and medical diagnosis can be carried out according to speaker's age [32].

Speech production from its inception at Esophagus to its delivery at the tip of tongue and lips passes through different acoustic changes. This anatomical region undergoes some changes throughout the life time of a person. The movement of the tongue, lips, jaws and other organs in the articulatory system produces sound. These organs create pressure which eventually leads to acoustic signals [33]. The movement of the organs is incredibly quick, delicate as it is controlled by brain and complex in its nature[34].

When we speak we push air out of our lungs all the way to our mouth via the vocal tract which basically involve throat heavily. Different sounds are produced through the movement or vibration of the vocal cords along with our tongue and lips which changes the air flow. A perceptible change in the sound we hear is possible with a slight change in the position and movements of the organs. Below are some of the

most important parts (organs) of this system. Figure 1.1 depicts the articulatory system after the discussion [35].

The lungs are part of the articulatory system where sound production begins. When we breathe, air moves in and out of these two bag-like organs in our chest. When we speak, our lungs push air up past the vocal cords and through the rest of the vocal tract, the space in the throat, mouth, and nose where sound is produced.

The vocal cords or vocal folds are two small membranes found in our throat which produce sound. When the vocal cords are stretched tight and close together, they vibrate rapidly more than 100 times per second. As a result, the sound that comes out is louder. At a relaxed state of the vocal cords, the sound that comes out of them is quieter, like a whisper. Pitch is affected by the vocal cords. It is a measure of how high or low the voice is at a particular instant of time; which fundamentally means high or low in the sense that a musical note is high or low; it does not mean a high or low volume or loudness. When the vocal cords are stretched out longer, the sound has a lower pitch. When they are shorter, the sound has a higher pitch. The space between the vocal cords is called the glottis. The vocal tract moves to change the shape and size of its opening. This movement helps to produce varieties of articulations in different languages.

The lips are involved in the production of numerous consonants or voiced sounds: /p/, /b/, /m/, /w/, /f/, and /v/. Certain ways of lip movements such as —making them rounded, unrounded, or stretched a bit wide—also affects the sounds of vowels.

The teeth are greatly engaged when we try to say the consonant sounds /f/ and /v/, with the upper teeth touching the lower lip, and also /θ/ and /ð/, with the tip of the tongue touching the upper teeth. These sounds are commonly known as fricative sounds in acoustics.

The alveolar ridge is the slightly rough area just behind the top teeth. It can also be called the tooth ridge or the gum ridge.

The tongue touches or almost touches the alveolar ridge when a speaker says the sounds /t/, /d/, /s/, /z/, /l/, and /n/. In addition with a collaborative effort with teeth it produces /th/ sound which is extremely hard for non-native English speakers. In fact the tongue is involved in producing almost all the sounds of English, both consonants

and vowels. We can also refer to different parts of the tongue: the tip of the tongue, the blade of the tongue, and the back of the tongue.

The hard palate is the hard part at the top of the mouth, beginning just behind the alveolar ridge. It can also be called the roof of the mouth. When we close our mouth, our tongue is probably flat against our hard palate. The tongue touches or almost touches the hard palate when we say the sounds /f/, /z/, /tʃ/, /dʒ/, and /y/.

The soft palate is the softer part of the roof of the mouth, farther back than the hard palate. It is also called the velum. If we touch the roof of our mouth with our tongue and then keep moving our tongue farther back, we will find that softer area. The back of the tongue touches the soft palate when we say the sounds /k/, /g/, and /ŋ/.

The nasal cavity is the space inside the nose where air passes in and out when we breathe through our nose. In some occasions it is referred to as the nasal passage. This area is important in producing the nasal sounds /m/, /n/, and /ŋ/ [36]. These sounds are especially important in speaker recognition. For these sounds, the air stream moves up and out through the nose instead of the mouth. The articulatory system organs that play roles in generation of speech signal are shown in Figure 1.1 below.

The mathematical model of speech generation is displayed in Figure 1.2 below. In this model, the cross-sectional area of the oral cavity $A(x, t)$, from the glottis, at $x = 0$, to the lips, at $x = L$, is determined by five parameters: tongue body height, anterior/posterior position of the tongue body, tongue tip height, mouth opening and pharyngeal opening. In addition, a sixth parameter is used to additively alter the nominal 17-cm vocal tract length [36].

The pressure created during a certain speech session $p(x, t)$, the volume velocity $u(x, t)$, the cross sectional area $A(x, t)$, position x and time t satisfy the following pair of partial differential equations given in (1.1) and (1.2) which basically express Newton's law and conservation of mass respectively. The symbol c is the speed of light in equation (1.2).

$$-\frac{\partial p}{\partial x} = \frac{\rho}{A(x, t)} \frac{\partial u}{\partial t} \quad (1.1)$$

$$-\frac{\partial u}{\partial x} = \frac{A(x,t)}{\rho c^2} \frac{\partial p}{\partial t} \quad (1.2)$$

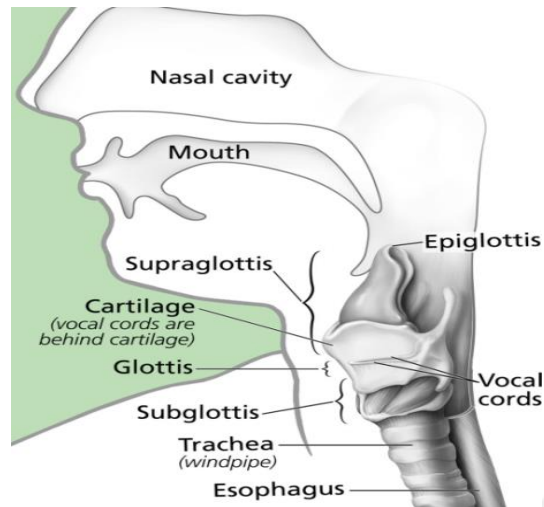


Figure 1.1. The vocal tract and the process of speech production

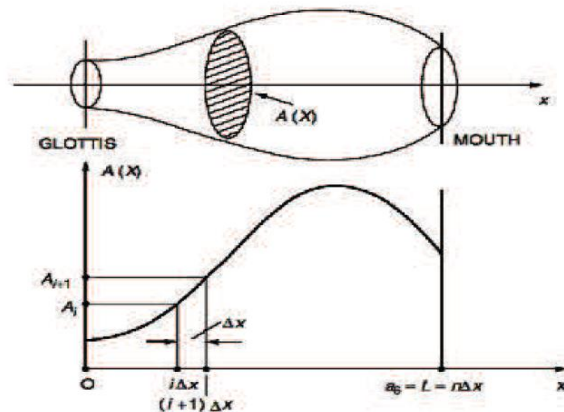


Figure 1.2. The acoustic tube model and the vocal tract area function

Each individual phoneme in speech production in any language can be categorized as voiced and non-voiced sounds. The non-voiced sounds in the English language consists of the sounds (a, e, I, o and u) commonly known as vowels whereas the voiced sounds consist of the majority of the consonants. These sounds have their own typical characteristics which makes them possible to identify during speech recognition problems. The speech wave is conceived at the far inner end of the vocal tract with a great deal of assistance by lung, diaphragm and other breathing system organs. It finally emerges at the outer end with the help of our lips, nose, tongue and teeth as an acoustic wave. So basically speech is a result of a series of vibrations due

to the pressure created during collision of organs in our articulatory system. The pattern of our speech goes through gradual changes as we get older because these organs undergo certain changes over the years we lived.

A simplified model of the vocal tract assumes the vocal passage as a tube of non-uniform and time varying cross section. As the air in this cavity varies in pressure it creates unique and distinct speech sounds. The vocal, glottal and radiation models consider soft walls, effect of friction and thermal conditions. The source and radiation models try to present the mathematical descriptions of the phenomenon that occur at glottis and lips during a speech session respectively. The glottal model is involved only to describe the voiced sounds. A random noise replaces the glottal transfer functions during unvoiced sounds. The glottal [37], vocal tract, radiation and the general speech models are given in equations (1.3), (1.4), (1.5), (1.6) and (1.7) respectively. In fact equations (1.3) and (1.4) are used to compute the glottal model.

$$g[n] = \begin{cases} 0.5 \left(1 - \cos \left(\frac{\pi n}{N_1} \right) \right) & 0 \leq n \leq N_1 \\ \cos \left(\frac{\pi(n-N_1)}{2N_2} \right) & N_1 \leq n \leq N_1 + N_2 \\ 0 & \text{otherwise} \end{cases} \quad (1.3)$$

This glottal approximation model is proposed by Rosenberg [36]. In the z-domain the glottal pulse model for voiced speech is approximated as

$$G(z) = \frac{1}{(1-z^{-1})^2} \quad (1.4)$$

Whereas, $G(z) = 1$ for unvoiced speech.

The shape and size of the vocal tract tube undergoes gradual changes. This non-stop change affects the models that are being discussed here. Age class and gender based models are necessary to address these changes. The nature of the tube shown in Figure 1.2 determines the characteristic feature of the speech uttered. It is not only the size and shape of the tube that affects the nature of the speech produced but also the inner surface of this tube matters much. It is not only the size and shape of the tube that affects the nature of the speech produced but also the inner surface of this tube matters much.

$$V(z) = \frac{G}{1 - \sum_{k=1}^N a_k z^{-k}} \quad (1.5)$$

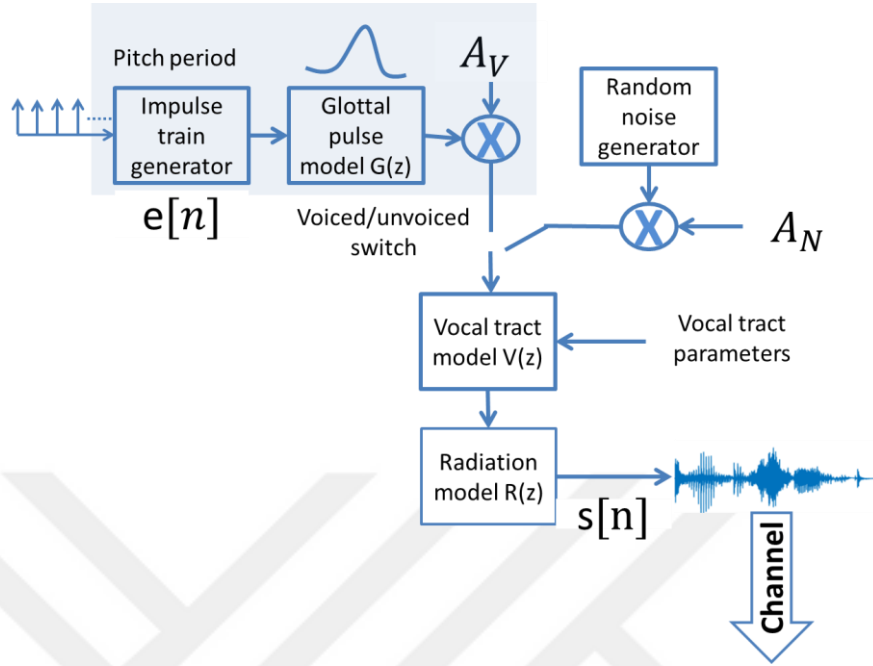


Figure 1.3. Speech production model

Equation (1.5) above approximates the majority of sounds with all-pole vocal tract model. The speech radiation at the outer end of the vocal cavity by the lips, teeth, tongue and nose is approximated using a transfer function with a zero slightly inside the unit circle as:

$$R(z) = 1 - \alpha z^{-1} \quad (1.6)$$

Typical values of α include 1 and 0.98 in equation (1.6). Finally the overall transfer function for speech production is given by

$$\frac{S(z)}{E(z)} = A_V G(z) V(z) R(z) \quad (1.7)$$

Where, $S(z)$ and $E(z)$ represent the produced speech wave and the initial excitation in z-domain respectively.

Our ear has got 3 sections namely; the outer, middle and inner ear. These sections constitute the auditory system. The perception process begins with filtering and converting the audio wave in to neural signal. Neural transduction is performed between the inner ear and the neural pathway to the brain. Recently a variety of

models that can simulate our auditory and perception capabilities are proposed. With the re-emergence of the neural networks, these models have considerably improved. Spectral signals coming from the medium, mainly the air (atmosphere) are converted in to neural activity signals in basilar membrane. Finally the neural activity is converted to language code in our brain

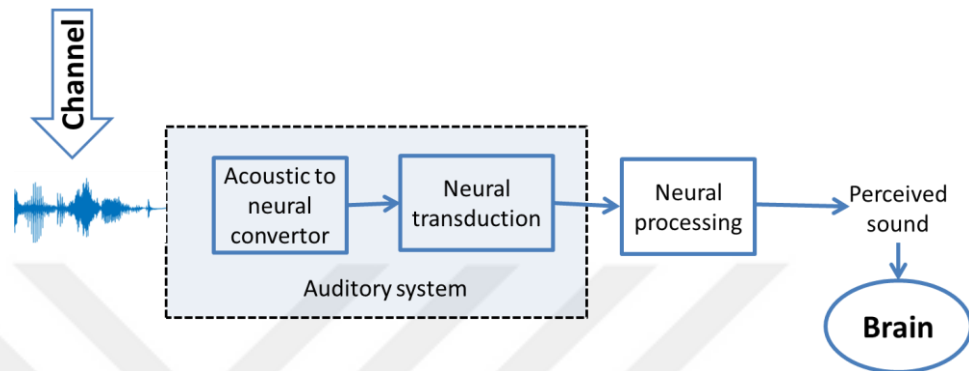


Figure 1.4. Speech perception process

Figure 1.5 below summarizes the whole synthesis to interpretation of speech. The origin of every speech is the human brain and its eventual destination is also brain where the original message is interpreted and understood.

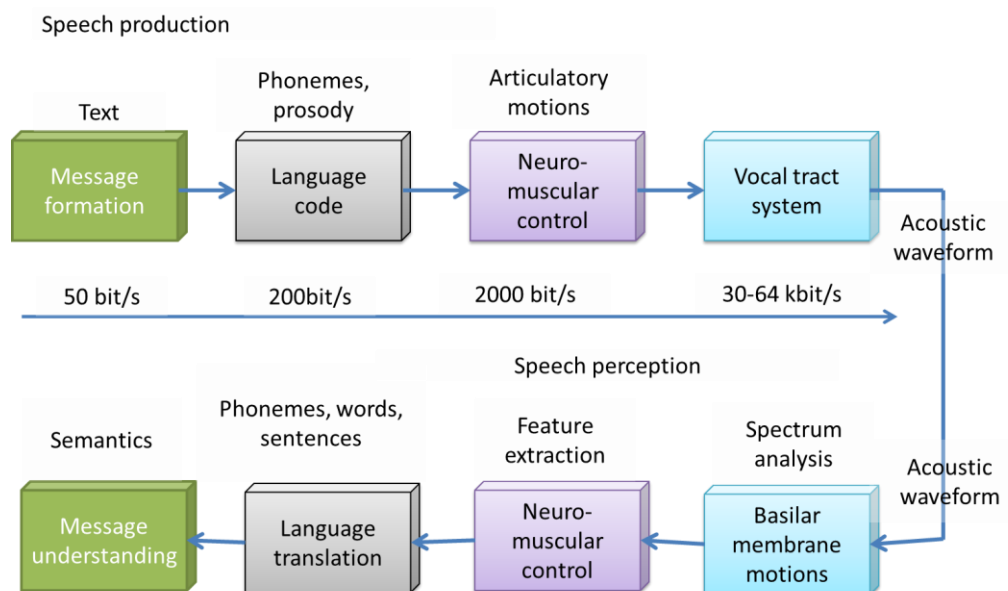


Figure 1.5. Inception, generation, propagation and interpretation of speech

The message conceived in the brain is converted to language codes. Prosody (syntax or rhythmic aspect of language), markers denoting duration of sounds, loudness and

itches are included in the codes. The next step is to initiate neuro-muscular commands to provoke the vocal cords in order to vibrate at suitable circumstances.

1.3. Procedural and Rule-based Classical Programming versus Artificial Intelligence

Classical and traditional programming paradigm needs inputs and a set of rules to act upon these inputs to produce intended outputs. But artificial intelligence makes it possible to train a machine to learn rules based on a huge amount of data attributes. The subclass of a machine learning discipline called Deep learning works much more like a human evolution [29]. When humans were few they had a very simple and uncivilized way of life. As time goes on the human population grows exponentially. As a result, the rules and ways of life started to be more complex than before. People started to learn new rules to make their life better and easier. Complex rules have been developed. Rules that have been accepted as the best were changed by new rules as humans learn new ways, new philosophies and new societies as well as new world. Similarly, a deep learning system creates poor rules over few data. But as our data grows, the system automatically gets better and better. Data is very important for a deep learning system to be all inclusive.

As we have tried to see the analogy between a human evolution through thousands of years and the machine learning process since its first invention, they have both shown a better progress in terms of simplicity and speed of task execution. This progress comes due to training with variety of data and better philosophy. For a human evolution, life philosophy could be a good aspect to explain the rules and procedures man has followed through the years. Whereas, computing machines exploit algorithms and/or methods to produce an output based on input attributes given to them.

Since the beginning of the second decade of this millennium researchers have been attracted to neural networks. Neural networks did not give good results during the first attempts. It was hopeless when the first perceptron was proposed to anticipate the human neuron. But some dedicated scientists have not given up on their study to find successful results. They kept on believing that a neural network eventually works out. Recently artificial neural networks connected in a successive manner have

been designed and applied to various classification problems and found to deliver amazing results. However a big question lies in every research that researchers could not give an answer to the question “why neural nets start giving amazing results?” In fact there is no question about how it works. Major part of our research incorporated deep neural networks (DNN) [29].

The diagram shown in Figure 1.6 below clearly explains the difference between classical programming and machine learning. Deep learning is a sub category of machine learning where initial weights of all synapsis connecting neurons of hidden layers as well as terminal layers gets updated every time a feature is given to the input layer based on the corresponding label. In Figure 1.6 rules basically stand for algorithms or a set of procedures based on which a machine gives an output. Data in our case stands for feature matrices or vectors extracted from speech frames and output stands for age classes or actual age values. The top block in this figure represents supervised learning in AI which needs labels to create suitable models. The PLDA and CDS in our study are among these algorithms.

A great many of AI algorithms however, train models without labels. These categories of models are said to be unsupervised learning algorithms. GMM is one of them. This class of algorithms also includes several deep learning models.

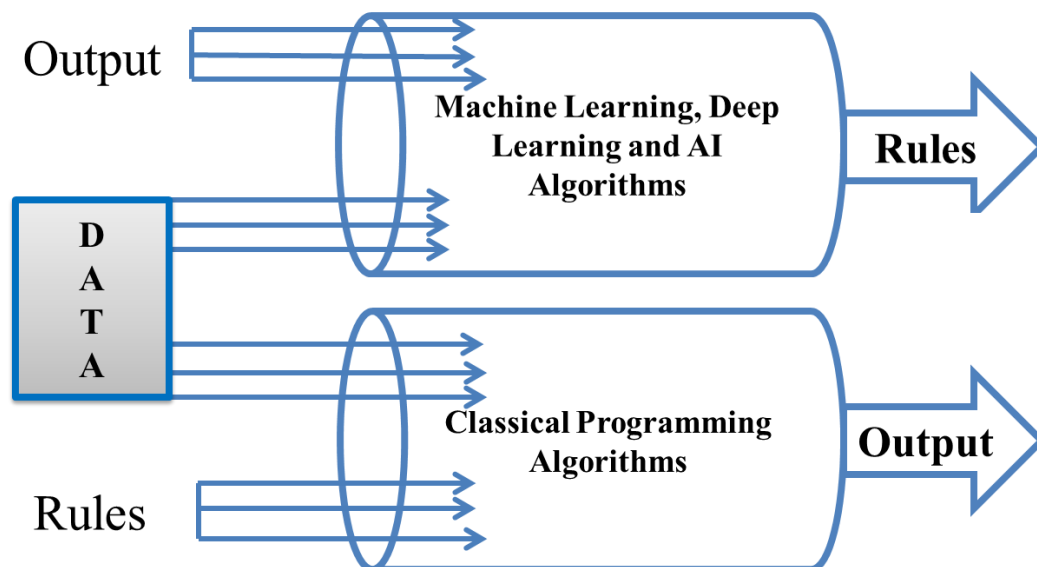


Figure 1.6. Classical programming versus machine learning paradigm

1.4. Literature Review and the Research Dynamics

Books, journal and conference articles, lecture notes, git-hubs, google forums, and internet resources are exhaustively used to proceed on this research study. The vast majority of our references are specifically related to speaker age estimation or classification. Among these literatures most of them also include gender detection or classification. Speaker recognition and verification studies come at the second top in our list of resources. Books on machine learning, pattern recognition, speech processing and programming paradigms are highly used in the process of this research.

We believe that looking at the brief history and developments of signal processing gives more insight in to speech processing and eventually brings us to speaker age estimation more specifically. It goes as far back as the 17th century, where we find the contribution of signal processing principles in the classical numerical analysis techniques according to Alan V. Oppenheim and Ronald W. Shafer renowned scholars in the signal processing discipline. In addition, digital control systems of the 1940s and 1950s consist of these principles in their operation according to Wikipedia sources. Speech is the most popular if not dominant among all signals which has been exploited, dealt, analysed, processed, transformed, made storable and used for communication more than any other signal. These operations made transmission and receiving audio signals through wired or wireless channels a lot easier. The half century old speech processing discipline was not and will not be all pretty easy and a free ride. It has always been challenging ever since its inception back in the 1950s. It went through several periods of intense promises. One of the most popular applications of speech processing; speech recognition began with the invention of Audrey, which is a digit recognizer, by bell Laboratories' researchers in 1952 [38]. The long and arduous desire for humans to design machines capable of mimicking human behaviors inspired researchers to devote their time and effort in the field of speech processing. The age old aspiration is to create a human-like machine able to recognize and synthesize speech. In its modern sense artificial intelligence played a vital role in many areas in the 21st century. AI emerged in the 1950s and has been being employed repeatedly in automating tasks otherwise performed by humans [29]. It was first coined at the Dartmouth conference in **1956**. It is a general discipline

which encompasses machine learning and deep learning. Symbolic AI, the dominant paradigm in AI from 1950s to the late 1980s, is having sufficiently large set of explicit rules and procedures for manipulating knowledge. Speech recognition, speaker recognition, speaker emotion recognition, accent recognition, speaker age estimation and many other speech research disciplines have been benefiting from AI subsets such as machine learning and deep learning.

The Latin phrase “annus mirabilis”¹ which means “marvelous year” originally used to refer the amazing works of Sir Isaac Newton for his laws of motion [39]. The year 1948 is widely regarded as the miraculous year (“annus mirabilis”) of signal processing with the emergence of a breakthrough research work entitled a mathematical theory of communication by Claude Shannon [40].

¹ “Annus mirabilis (pl. anni mirabiles) is a Latin phrase that means "marvellous year", "wonderful year", "miraculous year" or "amazing year". This term was originally used to refer to the year 1666 (of Isaac Newton), and today is used to refer to several years during which events of major importance are remembered. Prior to this, however, Thomas Dekker used the phrase mirabilis annus in his 1603 pamphlet The Wonderful Year.” as quoted from Wikipedia [39].

The initial sign of speaker recognition solely based on speech goes as far back as the biblical era in history where Isaac , who was unable to see because of old age, trying to recognize his two sons Esau and Jacob to give his blessings according to the book of genesis in the bible (Gen. 27:22-23) . The story tells a lot about age and identity information that exist in a human speech [41]. Speech similarity occasionally exists in family members of the same gender.

Speaker recognition and verification are undoubtedly the most widely dealt area of speech processing next to speech recognition. While the former is finding out who is speaking among many available candidates, the later compares a person’s speech with a given template [42]. Speaker recognition or otherwise known as identification is one to many mapping whereas speaker verification or authentication is a one to one mapping [43]. These two studies contain the speaker’s age information among others. While some depend on it, the majority of speaker recognition and verification researches do not relay on speaker age. Some are text independent whereas the majority still remains dependent on text.

The first attempt to deal with speaker age emerged in 1959 where the pitch and speech duration characteristics of older males are analyzed [44]. Three age groups; with average age of 47.9 years ranging from 32 to 62 years consisting of 15 adult individuals, with mean age of 73.3 years consisting of 12 elders ranging from 65 to 79 years designated as elder group I and with mean age of 85 years consisting of 12 senior individuals ranging from 80 to 92 years designated as elder group II are involved in this study. The study found out a rising mean fundamental frequency with age.

The British broadcasting corporation (BBC) in its “100 year life “section published an article entitled “The age you feel means more than your actual birth date” written by David Robson in 19th of July 2018 [45]. According to the article, most people feel younger or older than they really are and this feeling of subjective age has a big effect on their physical and mental health. This may also impact our speech patterns. It affects the way we speak psychologically. However, the actual age is unchangeable just like our height and shoe size. According to some scientists, subjective age could be the reason for some people to appear to flourish as they get older – while others fade. It has also been indicated in this article that, people become less extrovert and less open to new experiences. In conclusion, people tend to mellow as they get older according to the article.

Gender recognition is dealt in its own and along with speaker age in various studies. Results of a certain gender based study suggests that cross-gender acoustic differences are partly language dependent and could be socially constructed [31]. Gender recognition is a lot easier compared to speaker age recognition mainly due to the differences in average fundamental frequency (f_0), f_0 range, pitch period, phonation type and speech rate. For instance, the fundamental frequency (f_0) for children, female and male speakers, ranges 200-400, 150-200 and 50-200 Hz respectively.

Non-linguistic information such as speaker age can be extracted from speech signals using cognitive operations [46]. Speech rate can be considered as one source of information by which listeners use to extract speaker age information, particularly when listening to older speakers. Obviously, speech rate is not the only speaker age

tip, and when the speaker is relatively young. In spontaneous speech context listeners primarily relay on other sources of information such as acoustic and linguistic.

The first attempt to address the problem of age classification was made in the early 1950s [47], however this problem was supported by computer aided systems dealt based on information obtained from speech only recently[48]. Speakers of two databases, the Japanese speech corpus for large vocabulary (JNAS) and S(senior).JNAS, were divided into two groups by listening tests [48]. The speakers whose speech sounds so aged were put together. The other group has the remaining speakers of the two databases. After that, each: speaker group was modeled with GMM. Experiments of automatic identification of elderly speakers showed the correct identification rate of 91%. To improve the performance, two prosodic features were considered. These features are speech rate and local perturbation of power. The identification rate has been improved to 95% using these features. Using scores calculated by integrating GMMs with prosodic features, experiments to automatically estimate speakers' age have been carried out. Accordingly, high correlation between speakers' age estimated subjectively by humans and automatically calculated scores of 'agedness' was reported [47].

Acoustic feature sets were developed for speaker age estimation in a study conducted a decade ago [49]. MFCCs extended by a set of prosodic features, pitch, fundamental frequency, and first four formant frequencies are used as baseline feature sets. 220 features were obtained when these features are combined. Then, the 220 features are reduced by selecting the best feature subsets. Selection is done by maximizing the R² variance with R as correlation using multiple regression/correlation analysis. Eventually a mechanism is designed in their study to select the best subset composed of one feature, two features, and continues until there is no better subset. University of Florida Vocal Aging Database (UF-VAD) has been employed to test this approach. This database contains 5 hours of speech for 150 different speakers and 1350 utterances spoken in English. It has 3 age classes equally divided between males and females for young, middle-aged, and old elder. They generate a constant high-dimensional feature vector that is independent of the length of the utterance and of the extracted features for each speaker in the database and is represented by a

Gaussian model. Adding prosodic, pitch, and formant features to the MFCCs feature sets improved the results by reducing the mean absolute error between 4-20%.

Background noise, accent variation, speech duration, text-dependent or text-independent control variable, recording device variation, channel and space variability, and other related factors make speaker age classification as one of the most challenging tasks in speech processing research. Fusion of acoustic and prosodic level information offered weighted and unweighted accuracies of 49.5% and 52% respectively for speaker age classification. It also offered 88.4% and 85% accuracies for gender recognition likewise [50]. Speaker age classification consists of feature extraction and classification. A carefully designed feature extraction technique is not only able to extract age related features from the speech but also combats the effect of background noise as the noise coming from the surrounding is unavoidable. Classification in this context, is grouping training samples in discrete categories and to develop models for each category.

The modulation cepstrum coefficients instead of the cepstral coefficients for age and gender classification is proposed [51]. They extracted smooth information of the cepstral over a period of times for extracting frames from the speech utterance. The discrete cosine transform (DCT) was used over a fixed duration window. The speech utterance in modulation cepstrum domain has been filtered by decomposing the utterance cepstral trajectories into groups of low and slow frequencies. And then, the mel cepstral modulation spectrum (MCMS) features are extracted. The low modulation frequencies of MCMS (3-14 Hz) have the efficient information needed for age and gender classification as reported. A comparison of these features with the conventional MFCC was made and an accuracy of 50.2% using the MCMS features was reported.

Three novel systems which combine short-term and long-term cepstral features for speaker age recognition have been proposed and compared [52]. Pitches extracted from span of speech correlation clearly with the speaker age despite the fact that common successful systems such as GMM models and multiple phone recognizers that utilize such features have less performance than other features based on their acoustic analysis. Looking at independent performance of these two feature types,

short-term features are observed performing better than long-term ones with the feed forward DNN classifier in a certain study [53]. While a combined GMM/DNN classification scheme over short-term features offered 74.22% classification accuracy for female Turkish database, it showed more than 8% deficit for a DNN applied on long-term features with the same dataset according to this study.

In a 2016 speaker age estimation study, it was shown that the use of phonetically-aware i-vector extractor, could improve speaker age estimation performance compared with the GMM-UBM based counterpart [54]. Accordingly processing i-vectors through an LDA transform trained with discrete age labels dramatically sped-up the SVR training process in addition to improving speaker age estimation performance. DNN senone posterior based i-vectors method achieved speaker age estimation performance with a mean absolute error (MAE) of 4:7 years for both male and female speakers on the NIST SRE 2010 telephony test set. Basically the use of x-vectors for speaker age estimation is not the earliest development in speech research as x-vector embedding had been used for speaker verification (SV) before they were applied for speaker age estimation [55]. Robust speaker recognition was implemented using these state of the art embedding. The model was proposed by David Snyder for speaker verification (SV) [56] and later extended in 2018 [57]. Speaker verification project [55] exploited and has been built on top of kaldi recipe [58].

The question why estimating speaker age keep on being challenging remains one of the top areas of research in speech processing until an accurate or precise method is devised. Age estimation remotely has become more important than ever due to the emergence of violent and sensitive contents on the internet. These contents are unpleasant and harmful to children and young people. In addition it can be employed future technologies to settle possible tensions due to ageism.

The effect of aging on speech production patterns has been studied using two hundred Czech speakers whose age spans from 20 to 80 years old [59]. This study confirmed variations in temporal intensity and fundamental frequency domains across different age groups as well as genders. Based on the experiments carried on 200 Czech speakers, adult men are the fastest and most stable across utterances.

Supervised non-negative matrix factorization method is used for speaker age estimation and gender detection [18]. The method used hybrid architecture of weighted supervised non-negative matrix factorization (WSNMF) and general regression neural network (GRNN). Applying this approach on spontaneous read speech corpus in Dutch offered a mean absolute error rate of 7.48 years for age estimation and an accuracy of 96% for gender detection.

Information, such as speaker identity, gender, age range, and emotional state, are termed as paralinguistic information. Automatic recognition of this information can guide human computer interaction systems to automatically understand and adapt to different user needs. Several studies indicate that automatic age recognition could be a breakthrough in behavioral studies and health care as well. Much focus is given to the acoustic and prosodic level approaches for speaker age and gender identification [50]. Two baseline systems: Gaussian mixture model (GMM) on short-time spectrum based mel-frequency cepstral coefficient (MFCC) features, and support vector machine (SVM) on GMM mean super vectors have been considered in this study.

Scholars applied Utterance modeling with i-vectors to estimate speaker age [60]. This model has been used in conjunction with within-class covariance normalization (WCCN) and least square support vector regression (LSSVR) to address speaker age estimation which has achieved a Pearson correlation coefficient and mean absolute error of 0.772 and 6.08 respectively. Telephone utterances of NIST 2010 and 2008 are used for evaluation. The effect of some major factors influencing the proposed age estimation system, namely utterance length and spoken language are analysed in this scheme. Language, the communication channel at which the speech is recorded, and environmental conditions could affect the process of age estimation among other factors.

A comparison of human and machine estimation of speaker age conducted by Mark Huckvale and Aimee Webb showed that both human and machine automated approaches have difficulty in accurately predicting the age of elderly speakers [61]. The comparative study showed that human and machine accuracy is more similar with average errors of 9.8 and 8.6 years respectively. However the human estimation accuracy was believed to improve to 7.5 years if panels of listeners were consulted.

Both the age of speakers and listeners impacts the result. Children and young people do not have much experience compared to adults and older people to make relatively better estimation. Machines can also be thought analogously in a similar perspective as more data is fed to them they would improve their estimation capability. More experience observed in adults and elderly people is equivalent to more training data to machine estimation.

A mean absolute error (MAE) of 4.9, which is 14% better than the i-vector baseline, is achieved applying x-vector neural network architecture on NIST SRE08 dataset for training and NIST SRE10 for evaluation [62]. This architecture uses a series of time delay layers (TDNN) followed by a temporal pooling layer which summarizes the feature sequence into a single fixed dimension embedding. The embedding is fed into a series of feed-forward layers to predict the age value. The x-vector alone outperformed the i-vector baseline by 14%. In addition combining both the i-vector and x-vector improved the i-vector baseline result by 9%.

Support vector machine (SVM) is employed for speaker age estimation using the Gaussian radial basis function (RBF) as a kernel on MFCC and perceptual linear predictive (PLP) features as input sequences [63]. The gamma parameter on the RBF shows an improvement in speaker age estimation with smaller values but eventually starts degrading with a rise in gamma values. Speaker age estimation is proved to be better with 39 MFCC feature sets compared to 13 and 24.

Another interesting study on speaker age estimation includes a neural network back end used in an effort to replace classical classifiers and regression techniques which is carried out in 2015 [64]. The neural net is applied on i-vectors to generate speaker age values on test set speakers after the network is trained with a set of data reserved for training. According to this study carried out on national institute of standards and technology (NIST) database 2008 and 2010 conventional MFCCs with short term cepstral mean and variance normalization (CMVN) [65], worked best as the features for i-vector extraction; WCCN, and treating speakers as classes helped. However linear discriminant analysis (LDA) did not help considerably. To make it more understandable, no clear benefits were obtained with two-layer structure. The study suggested that a network with a single hidden layer trained with stochastic gradient

descent (SGD), is the recommended choice [66]. Eventually the artificial neural network (ANN) back end has reduced the MAE by 4.5% compared to support vector regression SVR [64]. Fedorova et. al indicated that the back-end may not have so much effect when the already compressed i-vectors are used as input features.

An attempt was made to use long short-term memory (LSTM) recurrent neural networks for speaker age estimation and explore its performance over 3 speech durations (3s, 5s and 10s) [67]. However, the emphasis was on backend mechanisms rather than the nature of the speech. LSTM is a neural network which has the ability of learning order dependence in sequence prediction studies. A similar attempt has been made to investigate effect of utterance length mismatch in training and test datasets using an end-to-end DNN approach [62].

A doctorate research carried out back in 2017 emphasized on generating new feature sets and deep neural network architectures for speaker age and gender classification [68]. Transformed mel-frequency cepstral coefficients (T-MFCC) are generated using DNN methods in [68]. This scheme has offered accuracies of 56.13%, 58.98%, 59.59% and 61.16% with T-MFCC with i-vectors as a class models based, T-MFCC with DNN as class models based, T-MFCC with DNN as speaker models based and fusion respectively.

It is reported that a certain DNN based classification experiment offered better speaker age and gender classification performance compared to traditional machine learning algorithms [69]. The researchers pointed out that age alone and a joint classification with gender offer different figures as 48.41% and 57.53% jointly with gender and age alone classification respectively. The gender classification performance is reportedly 88.8%. The joint gender and age classification is the most challenging task followed by age alone whereas gender classification is a lot easier compared to the other two tasks as the speech characteristics of males and females is distinct and separable.

Most recently, a multi-modal age corpus which can alleviate the challenges arising due to shortage of balanced and sufficient data has been established using the VoxCeleb2 database suitable for age estimation [21,70]. This database is used along with aGender and the Turkish database in our study [20]. It is reported in certain

studies that speaker age estimation is more challenging than facial age estimation.[21]. These studies also indicate that facial age estimation can be more robust. Another study which aims at estimating gender and age from speech signals applying state of the arts x-vector and transfer learning used Age-Vox-Celeb database [71].

Recently, age dependent insensitive loss has been used to estimate speaker age and short duration speech data has been employed for speaker profiling [72], [73]. The former study reported improvements in the mean absolute error (MAE) value ranging 3.1% to 5.2% using the NIST SRE 10 database as an evaluation set. And the later achieved MAE values of 5.2 years, and 5.6 years for male and female speakers respectively.

2. ADAPTED AND PROPOSED FEATURE EXTRACTION TECHNIQUES

2.1. Introduction

Speech is inherently regarded as a concatenation of discrete and finite set of symbols called phonemes. As stated in chapter 1 the main purpose of speech is communication. And the communication potential of speech can be characterized using the idea of the famous information theory proposed by Shannon [40]. Signal processing is obviously fundamental to feature extraction computations. The major part of it involves signal representation and transformation. Generally speaking, information processing and manipulation begins with identifying the source of information. Obviously a human speaker is our source when the information needed is embedded in speech. Therefore recorded audios organized as databases in gender and age classes are our source. Feature extraction is a distillation process. The values generated in this process, are believed to represent certain attributes of the signal. Figure 2.1 below shows general procedures in feature extraction operations.

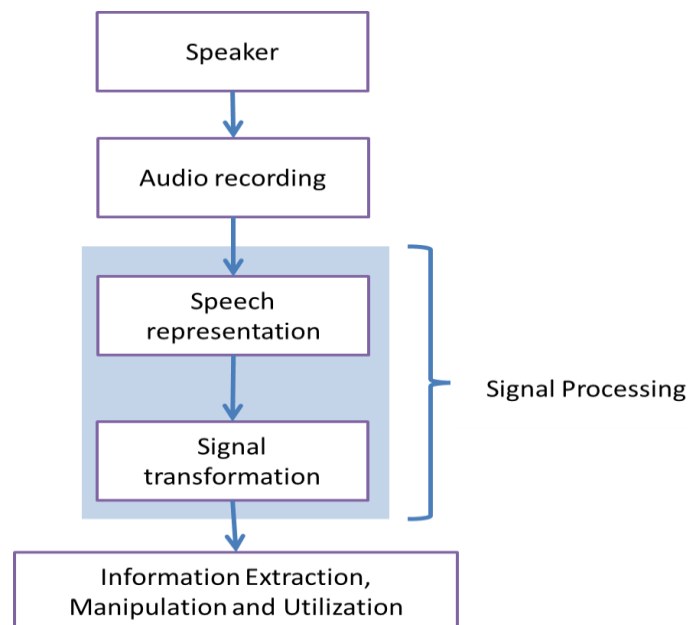


Figure 2.1. General block diagram of pre and post signal processing operations

The original representation size of the signal is reduced at the end of the process. A characteristic of large data sets is a large number of variables that require a lot of computing resources to process. Feature extraction is the name given for methods that select and/or combine variables into features, effectively reducing the amount of data that must be processed, while still accurately and completely describing the original data set. It reduces redundant values and focuses on unique values. For instance noise is a common characteristic of speech signals, therefore feature extraction works very hard to remove or reject it. But it should be noted that, losing important or relevant information must be avoided during the process.

The following subsection presents preliminaries to feature extraction; mainly time and frequency domain analysis of speech signal. This discussion will pave the way to the various feature extraction methods and operations as all of our experiments need these prerequisite computations.

2.2. Time and Frequency Domain Analysis of Speech Signal

2.2.1. Pre-emphasis

The majority of spectral energy of speech is concentrated at the lower end of spectral plots. At the higher frequencies however, the energy is much weaker. It is normally assumed that spectral energy roughly drops 2 dB for every 1 kHz of frequency increase (i.e. 2dB/kHz). This potentially causes practical problems in implementation. To compensate for such inaccuracies during implementation a pre-processing tool is required. A pre-emphasis finite impulse response (FIR) filter can play crucial role in amplifying spectral components at higher frequencies.

Excessive pre-emphasis however, would cause problems to fricative sounds as they have more energy at high frequencies. Therefore, the decision on how much pre-emphasis is needed depends on application and implementation details. Generally pre-emphasis filter serves to achieve the following objectives:

1. to amplify high frequency components
2. to balance the frequency spectrum
3. to avoid numerical problems during discrete Fourier transform (DFT) operations
and

4. to improve the signal to noise ratio (SNR) of speech utterances [74]

Given a discrete speech sequence $x[n]$ accessed using a Matlab command, $\{ [x, fs] = \text{audioread}(\text{wavFilePath}); \}$, the outcome signal $y[n]$ after applying pre-emphasis filter is defined as:

$$y[n] = x[n] - \alpha x[n - 1] \quad (2.1)$$

where the constant parameter α determines the cut-off frequency of the single-zero high pass filter through which $x[n]$ passes and usually assumed to be 0.94.

2.2.2. Windowing

The speech signal is extremely dynamic which changes its statistical properties within short period of time. For a stable and static analysis splitting up sentence level speech signals in to pieces good enough to represent a phoneme is needed. For non-stationary signals like speech spectral features in short segments rather than entire signal are of great importance for a great deal of applications.

Speech utterance of length L_s seconds sampled at f_s Hz contains $L_s * f_s$ number of samples. For instance a 3 second utterance sampled at 8 kHz is represented by 24000 discrete samples. This 3 second speech is not stationary in its statistic properties. Therefore cutting the 3 second speech in to smaller frames of length L_f seconds is carried out in order to get a new signal whose contents are capable of representing phonemes and maintain statistical properties stationary. The disadvantage of representing speech as a concatenation of independent and relatively stationary pieces of smaller frames is discontinuity. One way to avoid such discontinuity is by introducing overlap during framing. We shift the framing window so as to involve 25%, 50% or 75% of the previous frame which basically creates continuity between consecutive frames. We choose 50% in our experiments; however it can also be researchable parameter.

In the next subsections and entire unit we used $x[n]$ notation to represent a signal for a single frame, but to make it clear the full duration signal $X[n]$ is a superposition of all the frames in it. The following equation displays the mathematical description of speech using frames obtained through the process displayed by Figure 2.2 below

$$X[n] = \sum_{i=1}^{N_f} x_i[n] \quad (2.2)$$

The upper summation limit N_f represents number of frames which can be computed from the length or duration of speech L_s , length of frames L_f and overlap length or hop duration M as shown below.

$$N_f = \left\lfloor \frac{L_s - M}{L_f - M} \right\rfloor \quad (2.3)$$

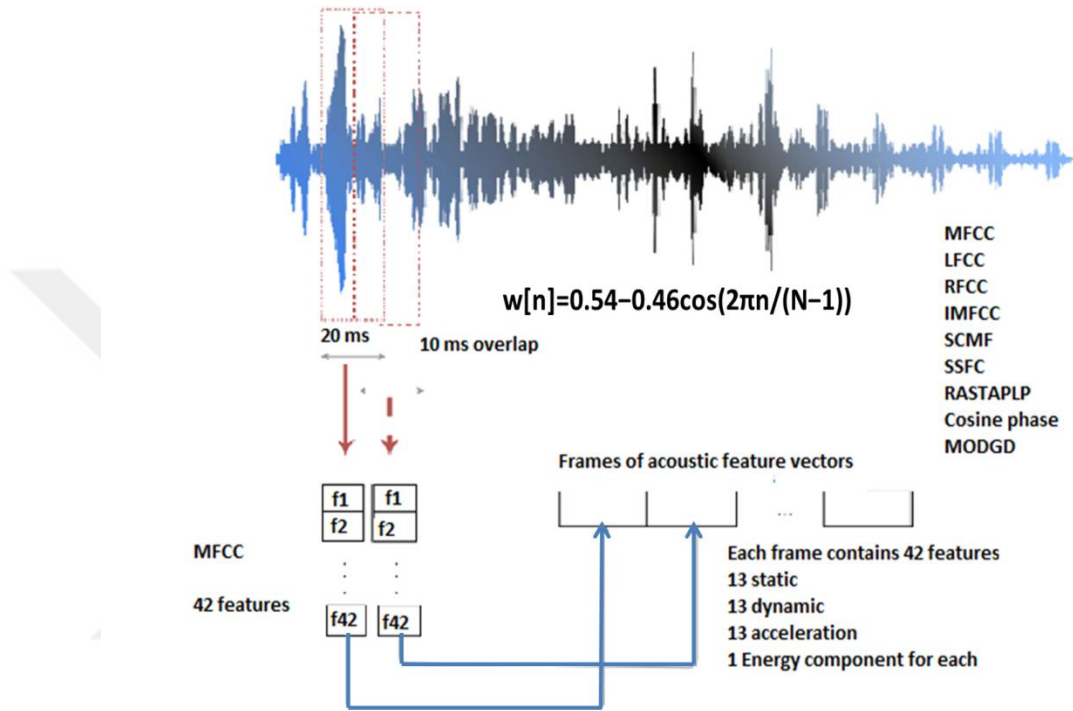


Figure 2.2. Framing and feature extraction using windows

A number of framing windows have been proposed for the purpose of splitting dynamic signals in to smaller and static enough for further processing. The four most popular windows in signal processing are rectangular, Hanning, Hamming and Bartlet whose discrete time functions are given in the following equations respectively [75].

$$w[n] = \begin{cases} 1, & 0 \leq n \leq L_f * f_s \\ 0, & otherwise \end{cases} \quad \text{where } M_f = L_f * f_s \quad (2.4)$$

Where the product, $L_f * f_s$, gives the number of discrete samples in a certain frame. Infact every frame consists of equal number of samples as the duration of every frame is equal.

$$w[n] = \begin{cases} 0.5 - 0.5\cos\left(\frac{2\pi n}{L_f * f_s}\right), & 0 \leq n \leq L_f * f_s \\ 0, & \textit{otherwise} \end{cases} \quad (2.5)$$

The Hamming window is the most commonly used window for the sake of reducing harmonics and leakage. Its function is given below

$$w[n] = \begin{cases} 0.54 - 0.46\cos\left(\frac{2\pi n}{L_f * f_s}\right), & 0 \leq n \leq L_f * f_s \\ 0, & \textit{otherwise} \end{cases} \quad (2.6)$$

Mathematical description of the Bartlett or triangular window is given below

$$w[n] = \begin{cases} \frac{2n}{L_f * f_s}, & 0 \leq n \leq \frac{L_f * f_s}{2} \\ 2 - \frac{2n}{L_f * f_s}, & \frac{L_f * f_s}{2} \leq n \leq L_f * f_s \\ 0, & \textit{otherwise} \end{cases} \quad (2.7)$$

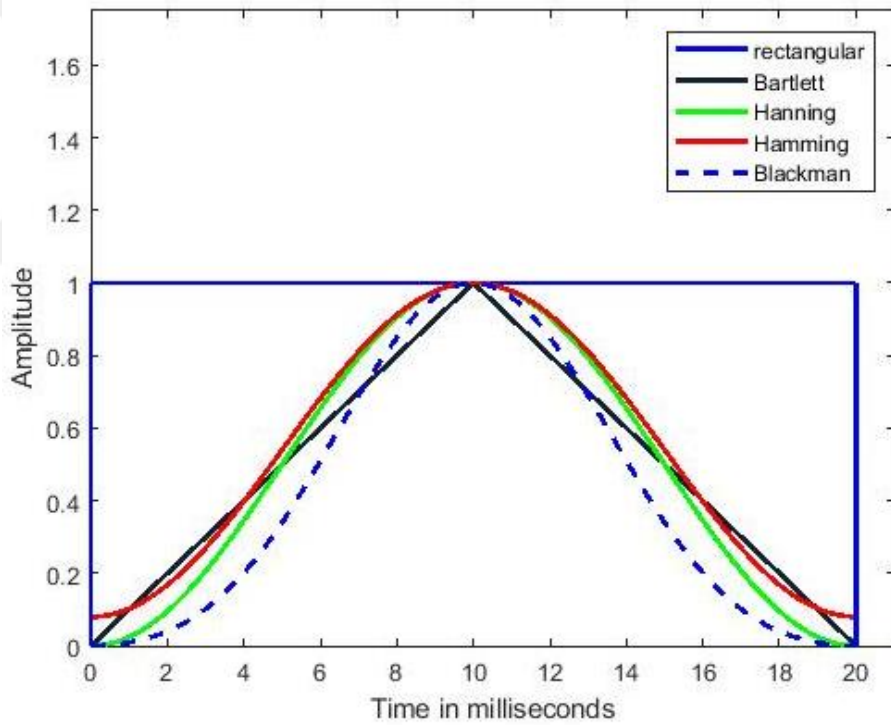


Figure 2.3. Framing windows

Finally the rarely used Black man window is given by

$$w[n] = \begin{cases} 0.42 - 0.5 \cos\left(\frac{2\pi n}{L_f * f_s}\right) + 0.08\cos\left(\frac{4\pi n}{L_f * f_s}\right), & 0 \leq n \leq L_f * f_s \\ 0, & \textit{otherwise} \end{cases} \quad (2.8)$$

The waveforms of the five framing windows discussed so far, are shown in Figure 2.3 above for a window length of 20 milliseconds.

2.2.3. Time domain analysis

The time domain analysis fundamentally assumes the speech signal $x(t)$ as dynamic and its properties change relatively slowly with time notably (5-10 sounds per second). It exhibits uncertainty due to small amount of data. Thus time domain processing of speech signal begins with a suitable representation and framing with one of the windows discussed in the above section to get stationary segment. There are basically two major choices for this task; waveform and parametric representation. The parametric speech representation is further classified as excitation and vocal tract parameters. The speech synthesis described in equation (2.1), is converted to analysis equation using product operation to get mathematical representation of each frame as shown in equation (2.9) below.

$$x_i[n] = X[n]w[n - (i - 1)\frac{M_f}{2}] \quad (2.9)$$

Where M_f represents number of samples in the framing window which can be computed as a product of the sampling frequency f_s and frame duration L_f ($M_f = f_s L_f$), $x_i[n]$ denotes the discrete time representation of the i^{th} frame, and the index i runs from 1 to the number of frames N_f in the utterance ($i = 1, 2, 3, \dots, N_f$).

Figure 2.4 below shows framing with a Hamming window of 50% overlap where every preceding frame consists of half unique and the other half similar contents with the current frame.

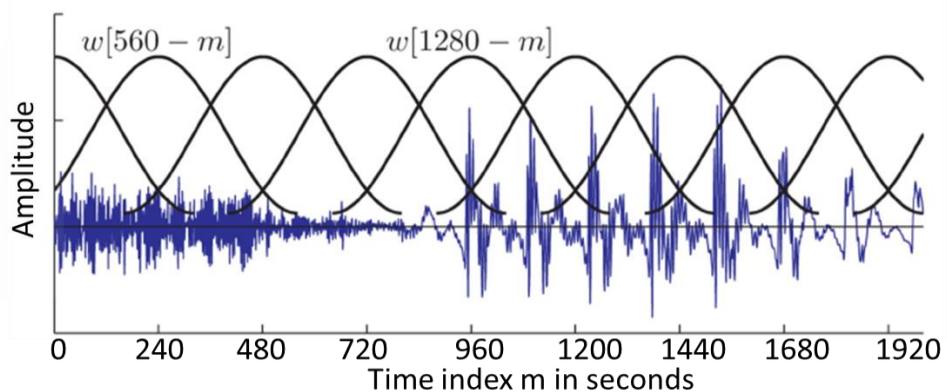


Figure 2.4. Framing with 50% overlap

Popular time domain speech computations following framing include:

- Zero crossing rate
- Level crossing rate
- Energy
- Autocorrelation
- Pitch range
- Average magnitude difference function (AMDF)

The zero crossing rate counts the number of sign changes for each sample in the entire frame. It is computed as:

$$Z_{CC} = \sum_{k=1}^N 0.5(\text{sign}(x[k]) - \text{sign}(x[k-1])) \quad (2.10)$$

It basically answers the question “how many times the speech signal crosses the time axis in a given frame. It is a reflection of frequency and high Z_{CC} value indicates high frequency.

Energy of a speech frame $x[n]$ is computed via adding all the squared samples in an entire frame as

$$E_s = \sum_{n=1}^N \{x[n]\}^2 \quad (2.11)$$

where N is the length of a certain speech frame $x[n]$. The short energy can also be computed from the frequency domain representation of the signal using Parseval’s theorem which will be shortly discussed in the next sub section.

Combined with short time energy can be used to detect voiced and unvoiced sounds as high energy E_s and low Z_{CC} values indicate voiced speech whereas low energy and high Z_{CC} values usually represent unvoiced ones. We recall that vibration of vocal cords is caused during voiced sound generation; on contrast unvoiced sounds do not need any vibration of the vocal cords. The voiced phonemes tend to be louder like the vowel sounds (/a/, /e/, /i/, /o/, /u/) whereas the unvoiced phonemes are abrupt like the sounds /p/, /t/ and /k/ [35]. These characteristics can be exploited in speech recognition applications.

Correlation function is commonly used in speech processing applications to show the difference between random variables. In quasi periodic signals such as speech we use

autocorrelation computation to uncover the distinction between a speech signal and its delayed version by k samples mathematically described as shown below

$$\phi(k) = \frac{1}{N} \sum_{n=0}^{N-1} x[n]x[n+k] \quad (2.12)$$

This equation is depicted in a block diagram shown in Figure 2.5. For a delay of k samples the average magnitude difference function (AMDF) which serves a similar purpose as autocorrelation function is given by

$$\phi(k) = \frac{1}{N} \sum_{n=0}^{N-1} |x[n] - x[n+k]| \quad (2.13)$$

Compared to autocorrelation AMDF may be less intensive to implement on some processor architectures.

Pitch period is another metrics which can be calculated in time domain as the inverse of the fundamental frequency of speech frame for voiced sounds. Pitch range is used

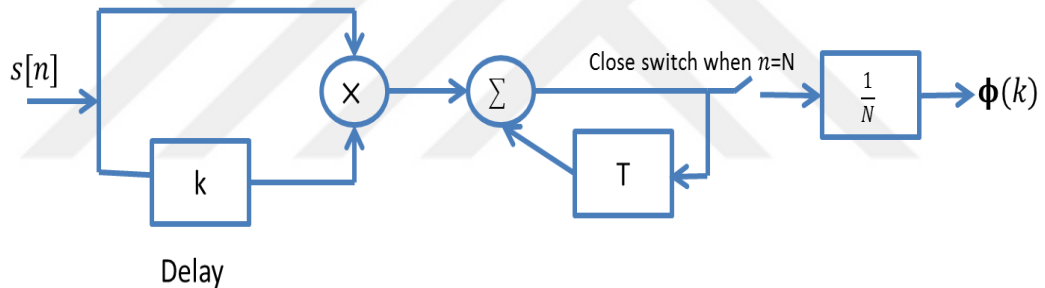


Figure 2.5. Autocorrelation function

2.2.4. Frequency domain analysis

It is fundamental to understand that all frequency domain analysis arise from the Fourier analysis of a certain signal. Fourier transform converts the time domain representation in to frequency domain in which we can visualize the magnitude and phase components of spectrums. For a discrete signal $x[n]$ the dual equations that compute frequency and time domain representations aka analysis and synthesis duo are given in equations (2.14) and (2.15) respectively.

$$X(e^{j\omega}) = \sum_{n=-\infty}^{\infty} x[n]e^{-j\omega n} \quad (2.14)$$

And the synthesis equation also called inverse Fourier transform is given as

$$x[n] = \frac{1}{2\pi} \int_{-\pi}^{\pi} X(e^{j\omega}) e^{j\omega n} d\omega \quad (2.15)$$

Equation (2.15) is commonly known as discrete time Fourier transform (DTFT). Properties of DTFT can be found in numerous academic resources [75]. The properties made complex computations a lot easier. Although the signal is discrete in time domain, it remains continuous in frequency domain which makes it difficult for digital hardware to further process. Hence we need to sample the frequency domain and represent the signal in finite samples in frequency domain too. Periodicity and symmetry are the most important properties. Since DTFT computation is angular and conterminal angles offer the same spectral value, unique values occur only for a unit circle whose angular range is $[0 \ 2\pi]$.

$$X[k] = \sum_{m=0}^{L-1} x[m] w[m] e^{-j\frac{2\pi}{N}km} \quad (2.16)$$

Where $w[m]$ is a framing window of length L and k is a frequency index which spans to a discrete Fourier transform (DFT) point N . Usually the DFT point is equal to the discrete sequence size L (i.e. $N = L$). In this sense, the spectral values are calculated only for discrete and finite angular frequencies given by

$$\omega_k = \frac{2\pi k}{N}.$$

$$\text{Hence: } X[k] = X(e^{j\omega}) \Big|_{\omega=\frac{2\pi k}{N}}.$$

If N is greater than L , zeros are padded to the discrete sequence $x[m]$. Whereas if N is less than L , $L - N$ samples that occur from time index N to L will be discarded from the sequence.

The DFT computation consists of a series of complex addition and multiplication operations. For an L -point DFT there are L complex multiplications and $L - 1$ complex additions for a single frequency index. Each complex multiplication consists of 2 real multiplication and 2 real additions. Therefore, a single frequency component of speech frame takes a total of $2[(L - 1) + L] = 4L - 2$ real mathematical operations. As we have a total of L frequency components for an L -point DFT, the total operation makes up $L(4L - 2)$ real operation which makes the complexity of the DFT computation to the order of L^2 and designated as $o(L^2)$. As the DFT point increases the complexity rises dramatically. Hence it makes it computationally inefficient to carry on these operations traditionally. Due to the

symmetric and periodic properties of the radix factor $e^{-j\frac{2\pi}{N}}$ efficient algorithms collectively termed as fast Fourier transform (FFT) have been proposed over the years. FFT reduces the computation complexity from $o(L^2)$ to $o(L * \log_2 L)$.

2.3. Filter Bank Based Features

After spectral analysis the next major step in most speech processing applications is applying filter banks to attenuate the components differently and recombine them into a modified version of the original signal. Filter banks are arrays of bandpass filters that split a certain spectrum of speech frame into multiple components, each one carrying a single frequency sub-band. They come in various shapes and the spacing between consecutive filters can be linear or mel scale. The mel scale is the most commonly used spacing as listeners judge the melody and loudness of sounds in a logarithmic scale rather than a linear fashion.

Magnitude and filter bank based spectral feature sets used in our research are; mel-frequency cepstral coefficient (MFCC), rectangular filter cepstral coefficient (RFCC), inverted MFCC (IMFCC) and linear frequency cepstral coefficient (LFCC). After carefully examining the performance of these feature sets we proposed a new technique called parabolic filter mel-frequency cepstral coefficient (PFMFCC) [76] to generate features and contributed for publication. RFCC offered an impressive performance for an experiment aimed at detecting replay or spoofing attack [77]. MFCC and PFMFCC use mel scale to split the range of frequencies between the minimum and the maximum while LFCC and RFCC use linear scales in our experiments.

Since the features considered here in this note are frequency domain features, the discussion so far is common for all the features. The DFT is the standing point for all the features. Some features use the magnitude and others use the phase component. Further procedures make each feature extraction technique unique from the other.

2.3.1. Mel-frequency cepstral coefficient (MFCC)

Filter bank based spectral feature extraction techniques including MFCC vary only in the choice of the filter bank shape and spacing between adjacent filters we use. Some

of these techniques use the mel scale whereas others use the linear scale. Linear scale means that the frequency bands are linearly divided. On the other hand, mel scale is a frequency scale commonly found in psychoacoustics, i.e. it reflects how our ear detects pitch. The filter banks are approximately linear below 0.5 kHz and approximately logarithmic above that. MFCC is one of these features. It is well known and widely used in the speech processing community. And it is believed that encouraging results have been obtained in using this feature. MFCC uses the Mel scale for linearly spacing the filter banks. Mel is a term taken from melody which supposedly is inspired by the human hearing or perception system. Since our auditory system uses a decibel or logarithmic scale, the above determined DFT either power spectrum or phase need to be redefined in a log scale. In addition to the filter bank spacing scales, we also categorize feature extraction techniques based on the type of filter banks used. Some use triangular and others use rectangular. MFCC uses Mel scale to space triangular filter banks.

The first step in MFCC feature extraction is to determine the short time Fourier transform (STFT) of speech signal. The STFT to be determined the sampled audio signal is first grouped in small overlapping frames of about 20ms size. This can be easily done using windowing techniques. The hamming window is chosen conventionally in most applications. Once framed, the STFT can efficiently be computed using the FFT algorithm.

Then, DFT values are grouped together in critical bands and weighted according to the triangular weighting function shown below. These bandwidths are constant for centre frequencies below 1 kHz and increase exponentially up to half the sampling rate.

Before the mathematical analysis of MFCC we need to explain about the Mel Scale which relates perceived frequency, or pitch, of a pure tone to its actual measured frequency. Humans are much better at discerning small changes in pitch at low frequencies than they are at high frequencies. Therefore the mel scale makes our features match more closely to what humans hear. This is basically motivated by the human perception mechanism which is done in a human cochlea. It doesn't perceive acoustic waves in a linear basis rather it uses a logarithmic scale in decibels. Below

is the formula for converting a conventional frequency measured in hertz to mel scale which is best expressed as human hearing scale:

$$Mel(f) = 1125 \ln\left(1 + \frac{f}{700}\right) \quad (2.17)$$

$$Mel^{-1}(m) = 700\left(\exp\left(\frac{m}{1125}\right) - 1\right)$$

$$mel(i) = 401.25 + 243.3740 * i \quad \text{for } i = 0, 1, 2, \dots, 10$$

$$mel(i) = 401.25, 622.50, 843.75, \quad \text{for } i = 0, i = 1 \text{ and } i = 2$$

$$mel(i) = 2392.49, 2613.74, 2834.99 \quad \text{for } i = 7, i = 8 \text{ and } i = 9$$

Obtaining all the mel scale points helps us to calculate the frequencies at which the filter bank functions begin and end. We use the inverse of equation (2.17) to compute the corner frequency values after partition in mel scale. Accordingly we have the following list of frequencies for the filter bank functions to be determined: $f(i) = 300, 517.33, 781.90, 1103.97, 1496.04, 1973.32, 2554.33, 3261.62, 4122.63, 5170.76, 6446.70, \text{ and } 8000$. At low frequency these functions are spaced closely while they are sparsely spaced at high frequency showing the same behaviour as human hearing nature.

The Major steps in MFCC and other filter bank based feature extraction implementation are [78]:

1. Apply pre-emphasis filter to the speech utterance before splitting it up in to pieces
2. The signal has to be framed in short frames of duration 20-30 ms.
3. Periodogram estimate of the power spectrum has to be calculated for each frame.
4. Apply the mel or linear filter bank to the power spectra, sum the energy in each filter.
5. The logarithm of all filter bank energies need to be calculated.
6. The Discrete Cosine Transform (DCT) or inverse FFT of the log filter bank energies has to be taken.
7. Only 2-13 DCT coefficients has to be kept, and the rest has to be discarded.

Following these procedures the delta MFCC which can also be called as dynamic feature and the delta delta MFCC or the acceleration feature can be calculated from the above 13 MFCC coefficients. Calculate the dynamic coefficients from the 13

static coefficients and similarly calculate the acceleration coefficients from the dynamic coefficients.

In order to apply the mel scale and generate the filter bank functions we need to determine the lowest and highest frequencies. Assuming a lowest frequency of 300 Hz and the highest frequency of 8000 Hz for instance, converting these measurements to mel scale based on equation (2.17) yields 401.25 mels and 2834.99 mels. The next step is dividing this range in to linearly spaced filter banks based on the desired number of filter banks which in this case is assumed to be 10 filter banks. The range will be $2834.99 - 401.25 = 2433.74$ and the linear spacing yields $2433.74/10 = 243.3740$.

Therefore, we need to define the filter bank function $V_i[k]$ that plays important roles in signal processing. They are used in many areas, such as speech and image compression, and processing. The main use of filter banks is to divide a speech frame in to several separate frequency domains. The triangular bandpass filter bank functions are mathematically defined as:

$$V_i[k] = \begin{cases} 0, & k < f(i-1) \\ \frac{k-f(i-1)}{f(i)-f(i-1)}, & f(i-1) < k < f(i) \\ \frac{f(i+1)-k}{f(i+1)-f(i)}, & f(i) < k < f(i+1) \\ 0, & k > f(i+1) \end{cases} \quad (2.18)$$

The corner points of each distinct triangle $f(i-1)$, $f(i)$ and $f(i+1)$ denote the lower, center and upper edge of the i^{th} filter bank respectively in equation (2.18). Once we determine these corner frequencies for filterbanks, the next step is converting these frequencies in to frequency bins k using the sampling frequency and the number of FFT points. For 512 – point FFT and 8 kHz sampling rate the bins will be computed as in equation (2.19).

$$k_i = \left\lfloor \frac{(N+1)*mel^{-1}(i)}{f_s} \right\rfloor \quad (2.19)$$

Here k_i is a frequency bin and i is the mel index at which we convert to frequency and eventually to bins.

$$MF[i] = \frac{1}{A_i} \sum_{k=L_i}^{U_i} |V_i[k]X(n, k)| \quad (2.20)$$

$$A_i = \sum_{k=L_i}^{U_i} |V_i[k]|^2 \quad (2.21)$$

$$MFCC[m] = \frac{1}{R} \sum_{i=1}^R \log(MF[i]) \cos \left[\frac{2\pi}{R} \left(i + \frac{1}{2} \right) m \right] \quad (2.22)$$

The filter bank energy is designated by A_i as in equation (2.21). In addition R and m stand for number of filter banks and number of features respectively in equation (2.22). While the m static features are extracted in the above procedures, the dynamic a.k.a. delta and acceleration a.k.a. delta-delta or double delta features can be computed using the following two equations. The m dynamic features are generated as in equation (2.23):

$$delta[t] = \frac{\sum_{n=1}^Q n(MFCC_{t+n} - MFCC_{t-n})}{2 \sum_{n=1}^Q n^2}, \quad Q = 2 \quad (2.23)$$

The m acceleration features are computed applying equation (2.24) shown below.

$$double_delta[t] = \frac{\sum_{n=1}^Q n(delta_{t+n} - delta_{t-n})}{2 \sum_{n=1}^Q n^2}, \quad Q = 2 \quad (2.24)$$

All the mathematical computations presented above equally apply for the other feature extraction techniques with a modification on the bandpass filter bank functions according to their shape and spacing.

2.3.2. Rectangular filter cepstral coefficient (RFCC)

This feature is similar with MFCC in its filter bank spacing scale. Both use the Mel scale. However RFCC as its name implies uses rectangular filter banks before calculating the cepstral coefficients. And the filter banks are computed using trapezoidal membership function. The Matlab inbuilt function *trapmf(x, parameter)* is used to determine the filter channels where x defines the domain and ‘parameter’ assigns the corner values of the trapezoid. Since we have four corners in a trapezoid a one by four array need to be assigned to the parameter.

```
for i=1:M
    fft_matris(i,:)=trapmf(fft_fr,[F_mel(i),F_mel(i),...
        F_mel(i+2),F_mel(i+2)]);
End
```

This piece of code fragment constructs M filters spaced regularly in a Mel scale. This way it computes values for all the rows in an iterative way and accumulates in the *fft_matrix(i,:)* matrix. All procedures after this code fragment remain the same as MFCC method

2.3.3. Linear frequency cepstral coefficient (LFCC)

Triangular filter banks are employed both in MFCC and LFCC however; LFCC uses linear scale frequency spacing as opposed to the Mel scale in MFCC. The Matlab code fragment to compute the M filter banks is given below. It is exactly the same as the MFCC code fragment. The difference lies in the values of the lower, center and upper frequency values in each filter bank. All these corner frequencies are spaced linearly in LFCC where as a mel scale is used in MFCC.

Algorithm 2.1:

```

for i=1:M
    fft_matrix(i,:) = ...
    (fft_fr > lower(i) & fft_fr <= center(i)).* ...
    filt_height(i).*(fft_fr-lower(i))/(center(i)-lower(i)) + ...
    (fft_fr > center(i) & fft_fr < upper(i)).* ...
    filt_height(i).*(upper(i)-fft_fr)/(upper(i)-center(i));
end

```

Where lower(i), center(i) and upper(i) stand for lower, center and upper frequencies respectively. In the linear filter cepstral coefficient (LFCC) we take the minimum and maximum frequencies and divide the length in to M equal length small segments. In the MFCC case the minimum and maximum frequencies are converted in to Mel scale first and we use these values to divide the length in to M small segments. The conversion is shown below.

Algorithm 2.2:

```

F_max_mel=(1000/log10(2))*log10(1+F_max/1000);
F_min_mel=(1000/log10(2))*log10(1+F_min/1000);
F_mel=linspace(F_min_mel,F_max_mel,M+2);

```

Then we need to convert these individual corner frequencies for all the M filters back to frequency scale (Hz).

Algorithm 2.3:

```
F_Hz=1000*(-1+10.^(F_mel.*log10(2)/1000));  
lower=F_Hz(1:M);  
upper=F_Hz(3:M+2) ;  
center=F_Hz(2:M+1);
```

Note that, LFCC does not need mel conversion as its scale is already linear.

2.3.4. Inverted mel-frequency cepstral coefficient (IMFCC)

This technique is also similar with MFCC in most procedures. It uses triangular filters and mel-frequency scaling to space the filter banks just like MFCC does. But the filter banks are inverted. To make it clear, the filter banks are narrowly spaced at low frequency and the spacing gets longer and longer as frequency increases in MFCC whereas the spacing length gets shorter and shorter as frequency increases in IMFCC. After calculating the filter impulse response functions (fft_matrix) in a similar way as MFCC using code fragments shown above, these filter bank impulse response values will be flipped using `fliplr()`.

```
LFBE=10*log10((fliplr(fft_matrix) * spectrum)+eps);
```

2.3.5. Parabolic filter mel-frequency cepstral coefficient (PFMFCC)

Inspired by all the four filter bank arrangements discussed above, we proposed a similar type of filter banks but parabolic in shape to get a different result. Every individual passband filter except the first one begins rising from the center of its previous filter. The center of the filter bank is the point at which the impulse response scores is maximum value.

We made a unique contribution at the filter bank stage in this study which resulted in a different set of features. The most commonly used triangular band pass filter banks in MFCC are replaced by parabolic filter banks inverted down and shifted to the right based on mel scaling of range of frequencies. The general description of the i^{th} filter bank function $H[i, k]$ is given in a compact form as in equation (2.25):

$$V_i[k] = -A(i)(k - f(i))^2 + B \quad (2.25)$$

In order to obtain the maximum and the vertical line of symmetry of this function a first derivative with respect to k must be applied and equated to zero. This leads to the equation given in (2.26):

$$(V_i[k])' = -2A(i)(k - f(i)) = 0 \quad (2.26)$$

which in turn leads to the point where the maximum of the function occurs. The vertical line $k = f(i)$ is the line of symmetry at the same time the maximum value of the i^{th} parabolic function occurs here. And the maximum point is $(f(i), B)$. The intercepts to the horizontal axis are determined based on the values of the partition made on the frequency range (f_{min}, f_{max}) . The minimum frequency is set to be 0 and the maximum is half of the sampling frequency used in the speech database which can be written as $(f_{min}, f_{max}) = (0, 4000\text{Hz})$ for the aGender database [20]. Whereas $(f_{min}, f_{max}) = (0, 8000\text{Hz})$ is used for the Turkish database as the utterances are sampled at 16 kHz.

The intercept values determine the parameter $A(i)$ in each parabolic function. The value of the function below the minimum and beyond the maximum intercept should be set to zero. The maximum value B remains the same in all the band pass filter bank functions. The number of filter bank functions is set to be 30 in our experiments. Once the entire range of frequencies is converted in to mel scale using equation (2.17), this range is partitioned in to 30 smaller band of frequencies. It is known that the mel scale relates the perceived frequency to the actual measured frequency. The human ear is better at identifying small changes in speech at lower frequencies. The converted minimum and maximum frequency pair in mel scale is $(0, 2146)$.

The filter bank functions in (2.25) above need to be redefined considering the intercept points of the functions at the corner frequencies. The value of each filter bank function is made to vanish out of the ranges of the intercepts. Substituting the right edge of each filter $f(i + 1)$ in the values of k and equating it with zero results in $A(i) = (f(i + 1) - f(i))^{-2}$, which eventually give the relation described in (10).

Since the parabola in each function is symmetrical with the line $k = f(i)$, the distance from the center to both edges is equal.

$$V_i[k] = \begin{cases} -\left(\frac{k-f(i)}{f(i+1)-f(i)}\right)^2 + 1, & f(i-1) \leq k \leq f(i+1) \quad , i = 1, 2, 3, \dots, 30 \\ 0, & \text{otherwise} \end{cases} \quad (2.27)$$

Where the left most edge $f(0)$ and the right most edge $f(31)$ of all the filter banks are 0 and 4000 Hz respectively in equation (2.27). The conventional triangular filter bank functions $V_i[k]$ employed for MFCC given in equation (2.18) above differ from the parabolic ones in having a sharp corner at the center of each function as a result each filter needs two line functions to define the entire sub band. The graphs of both triangular and parabolic filter banks are shown in Figures 2.6 and 2.7 respectively. In both cases the spacing is uniform until the thousandth frequency, and then it increases in each succeeding filter bank.

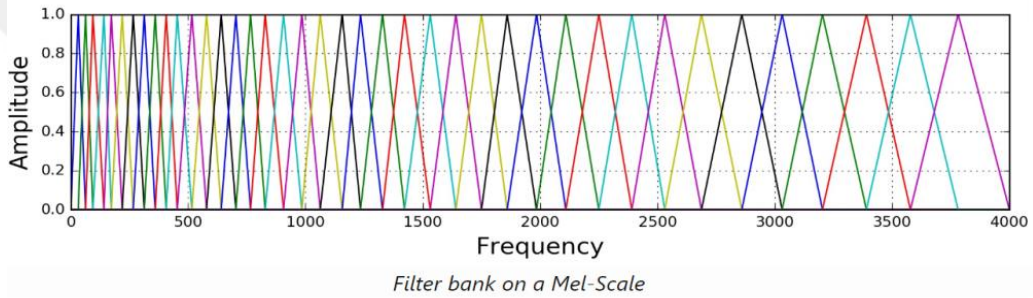


Figure 2.6. Triangular filter banks for MFCC

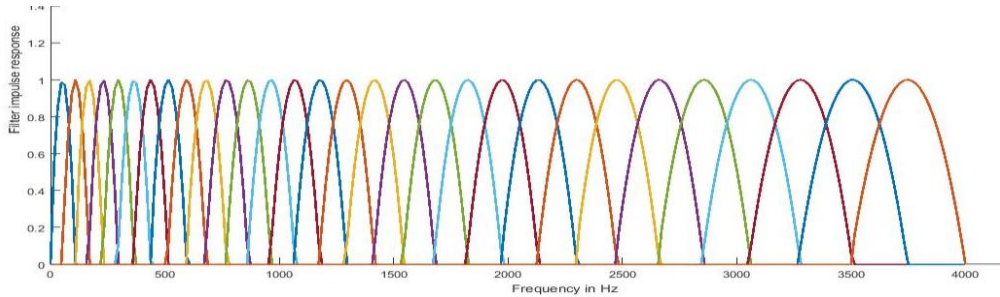
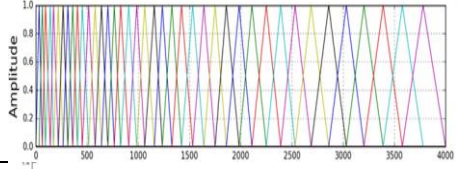
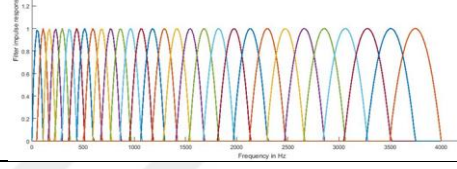
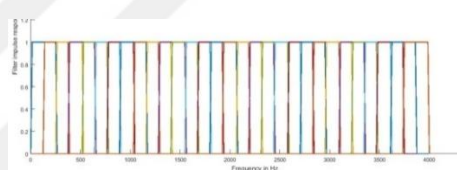
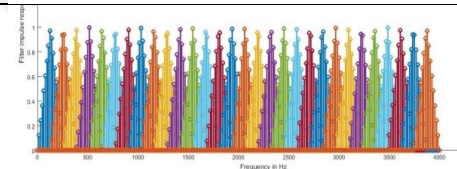
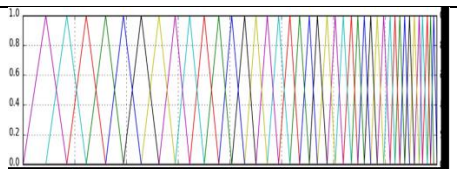


Figure 2.7. Parabolic filter banks for PFMFCC

Table 2.1 below summarizes the computation of filter bank functions presented so far in this sub section. In linear scale uniform partitions are made while splitting the frequency band between the minimum and maximum frequencies. Whereas, the mel scale uses logarithmic scale. First the maximum and minimum corner frequencies are converted in to mel scale and these values are used to determine the corner frequencies of each partition. Finally the corner frequencies of each partition are converted back to actual frequency in linear scale. At low frequencies below 1000

Hz the partitions in linear as well as mel scale are uniform. After 1000 Hz the mel scale partition starts increasing as the frequency increases.

Table 2.1. List of filter banks in magnitude spectral feature extractions

Features	Filter banks used
<p>MFCC</p> $V_i[k] = \begin{cases} 0, & k < f(i-1) \\ \frac{k-f(i-1)}{f(i)-f(i-1)}, & f(i-1) < k < f(i) \\ \frac{f(i+1)-k}{f(i+1)-f(i)}, & f(i) < k < f(i+1) \\ 0, & k > f(i+1) \end{cases}$	
<p>PFMFCC</p> $V_i[k] = \begin{cases} -\left(\frac{k-f(i)}{f(i+1)-f(i)}\right)^2 + 1, & f(i-1) \leq k \leq f(i+1) \\ 0, & \text{otherwise} \end{cases}$	
<p>RFCC</p> <p>We used trapezoid functions</p> $\text{fft_matris}(i,j) = \text{trapmf}(\text{fft_fr}(j), [F_mel(i), F_mel(i), F_mel(i+2), F_mel(i+2)]);$	
<p>LFCC</p> <p>Functions are the same as MFCC but the spacing is linear rather than mel scale as in MFCC.</p>	
<p>IMFCC</p> $V_{i_IMFCC}[k] = \text{flip}(V_{i_MFCC}[k])$ $\text{fft_matris} = \text{fliplr}(\text{fft_matris})$	

2.4. Phase and Sub-channel Based Features

In this sub section phase and sub-channel based as well as the old relative spectral transform-perceptual linear prediction (RASTA-PLP) are presented briefly [13].

2.4.1. Sub-band spectral flux coefficients (SSFC)

Spectral flux measures the spectral change between two successive frames and is computed as the squared difference between the normalized magnitudes of the spectra of the two successive short-term windows is given by equation (2.28) as:

$$Fl_{(i,i-1)} = \sum_{k=1}^L (E_i(k) - E_{i-1}(k))^2 \quad (2.28)$$

where $(E_i(k))$ is the k^{th} normalized DFT coefficient at the i^{th} frame which is given by equation (2.28) above.

$$E_i(k) = \frac{X_i(k)}{\sum_{j=1}^L X_i(j)} \quad (2.29)$$

And here $X_i(k)$ denotes the k^{th} DFT of frame i , and L is half of the DFT point N in equation (2.29). $\text{Spectral}_{\text{Flux}} = |\text{Normalized}_{\text{Spectrum}} - \text{Previous}_{\text{Spectrum}}|^2$ Spectral flux is the average variation value of spectrum between the adjacent two frames computed as:

$$Sf = \frac{1}{(N-1)(K-1)} \sum_{n=1}^{N-1} \sum_{k=1}^{K-1} [\log(X(n, k) + \delta) - \log(X(n-1, k) + \delta)]^2 \quad (2.30)$$

where Sf is spectral flux, $X(n, k)$ is the discrete Fourier transform (DFT) of the n^{th} frame of speech signal $x[m]$ in equation (2.30), N is the total number of frames, K is the order of DFT and δ is a very small value to avoid calculation overflow in equation (2.30) above [79].

2.4.2. Sub-band centroid magnitude and frequency (SCMF)

SCMF combines spectral centroid magnitude (SCM) and spectral centroid frequency (SCF) [80]. Given the frequency spectrum of a speech frame $x[n]$ as its Fourier transform $X(\omega)$ we divide the spectrum into M sub-bands. Each sub-band consists of a filter frequency response of $H_k(\omega)$ with a lower frequency f_{lk} and an upper frequency f_{uk} both the centroid magnitude and frequency are computed using (2.31) and (2.32) respectively as:

$$m_k = \frac{\sum_{f=f_{lk}}^{f_{uk}} f X(f) H_k(f)}{\sum_{f=f_{lk}}^{f_{uk}} f} \quad (2.31)$$

$$F_k = \frac{\sum_{f=f_{lk}}^{f_{uk}} f X(f) H_k(f)}{\sum_{f=f_{lk}}^{f_{uk}} X(f) H_k(f)} \quad (2.32)$$

where m_k and F_k denote centroid magnitude and centroid frequency.

2.4.3. Relative spectral transform perceptual linear prediction (RASTA-PLP)

Spectral transform (RASTA) is a separate technique that applies a band-pass filter to the energy in each frequency sub-band in order to smooth over short-term noise variations and to remove any constant offset resulting from static spectral coloration

in the speech channel e.g. from a telephone line [13]. RASTA-PLP achieved 98% and 95% gender classification accuracy for clean and noisy speech using robust GMM classifiers [81]. RASTA method of speech processing is a generalization of cepstral mean subtraction (CMS). The RASTA algorithm uses auditory masking principle in reducing the perception of noise. It addresses the problem of a slowly time-varying linear channel (i.e., convolutional distortion) in contrast to the time invariant channel removed by CMS. The essence of RASTA is a cepstral lifter that removes low and high modulation frequencies and not simply the DC component, as does CMS. The fixed infinite impulse response (IIR) band pass filter for all time trajectories given by transfer function in equation (2.33) is used by RASTA for noise reduction.

$$H(z) = 0.1z^{-4} \frac{2+z^{-1}-z^{-3}-z^{-4}}{1-0.94z^{-1}} \quad (2.33)$$

2.4.4. Cosine phase

Cosine phase is one of the two phase spectrum based features used in our research work during the speaker age classification experiments. It is extracted from the phase envelope of speech frames. phase spectrum information is normally ignored in most applications [82]. It is used in our study for age classification for the first time and found to perform poorly compared to most of the other features used in our experiments. The base phase feature $\zeta_{t,t-\Delta\tau}[k]$ for frequency channel k due to the phase dependence interference introduced by the cosine of the phase difference between two signals $\varphi_t[k]$ and $\varphi_{t-\Delta\tau}[k]$ at two different times t and $t - \Delta\tau$ respectively is given by equation (2.34) below

$$\zeta_{t,t-\Delta\tau}[k] = \cos(\varphi_t[k] - \varphi_{t-\Delta\tau}[k] - \frac{2\pi k}{N} \eta \Delta\tau + \theta_k) \quad (2.34)$$

where θ_k a phase shift created by feedback loops.

2.4.5. Modified group delay (MODGD)

The MODGD feature is the negative rate of change of the phase spectrum $\theta(\omega)$ with respect to frequency ω as defined in equation (2.35) below. This feature is used in [83] for speech recognition. It is used for speaker age classification for the first time in this study and the best result is achieved in the female test set.

$$\tau(\omega) = -\frac{d(\theta(\omega))}{d\omega} \quad (2.35)$$

$\theta(\omega)$ is taken from $X(\omega)$ written in its magnitude and phase components using polar representation as $|X(\omega)|e^{j\theta(\omega)}$

2.5. Feature selection, Feature Fusion and Dimensionality Reduction

Simple models are only able to describe data with fewer dimensions in a robust manner. As dimension increases the complexity emerges to be the curse of obtaining a feasible model to fully explain what underlies in data. Therefore, we need more rigorous methods to uncover knowledge from data. Contrary to this if we can explain certain data with less number of features; it is a lot easier to understand what underlies within data. Hence knowledge can easily be extracted from such data. Data can also be plotted and analyzed visually if it can be represented with fewer features such as hidden and latent factors without loss of information. Both principal component analysis (PCA) and linear discriminant analysis (LDA) are linear projections mainly used in dimensionality reduction processes however they are unsupervised and supervised respectively. PCA resembles much very much like factor analysis and multidimensional scaling.

For a 2 second speech, sampled at a sampling rate of 8 kHz the number of discrete values representing it would make up 16000 real numbers. It contains speech phonemes, noise and silence of course. The silence and noise contribute nothing if not negatively impact speaker age estimation efforts. With one of the methods discussed so far in this unit this dimension can be reduced to a much lower dimension assuming decompositions can be constant statistical characteristics. This 2 second speech will be decomposed in two 200 frames each containing 160 samples.

The number of samples in a frame multiplied by number of frames ($160 * 200 = 32000$) does not fit with the original 16000 samples as there is 50% overlaps between adjacent frames. This would even make the dimension higher. However we carry out a series of operations to reduce the 160 samples in to 42, 28, 14, 39, 26, or 13; depending on our choice of feature types, as static, dynamic, or acceleration; with or without energy components. If we choose the highest dimension here i.e. 42, 200

frames would make up 8400 discrete sequences which is a reduction of 42.5% from the original 16000 discrete sequences.

This reduction dramatically reduces the complexity of operation assuming the huge size of typical data. Therefore feature extraction is one of the dimensionality reduction methods along with feature selection [84], subset selection and others which reduces a d dimensional data to k where $k < d$ and discard the irrelevant $d - k$ dimensions.

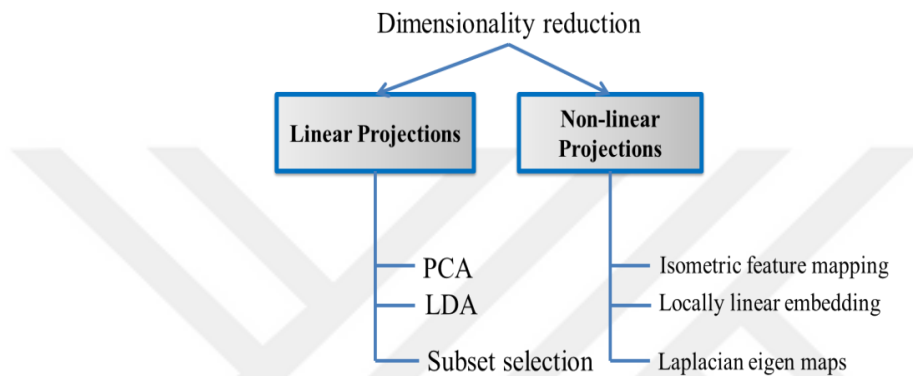


Figure 2.8. Dimensionality reduction methods

Figure 2.8 above depicts some linear and non-linear dimensionality reduction methods. Details on these dimensionality reduction methods and other search algorithms can be obtained in Ethem Alpaydin’s introduction to machine learning book [85].

We applied PCA, LDA, and subset feature type selection methods in our study. We adapted 2 phase and 7 magnitude based spectral features to our research study and proposed parabolic filter based feature extraction method which offered best results with certain classifiers. Combined effect of these ten features is investigated in this study. The following subsections present these dimensionality reduction methods briefly.

2.5.1. Union selection of feature sets

Subset selection is the process of finding the best subset among the set of features usually conducted in forward or backward search fashion using greedy algorithm [86]. The forward search begins with empty set and adds features to the subset depending on their performance, minimum mean absolute error (MAE) for

regression or maximum accuracy for classification. The process continues until there is no more improvement in performance or the improvement is insignificant. On the other hand, backward search starts with the complete set and removes the feature that gave the poorest performance until the improvement saturates.

We used the subset selection method in our experiments in a different approach. Rather than a single feature, we considered feature set types as we applied 10 kinds of feature sets. Before we make a subset of these feature types their performance is evaluated for three machine learning models; cosine distance scoring (CDS), Gaussian mixture model (GMM) [87] and probabilistic linear discriminant analysis (PLDA). Each feature type consists of 42 features. Therefore, a tradeoff is needed to minimize the cost of complexity as we perform subset selection of these feature types. Unless a significant improvement is made, feature types will not be padded to the existing subset in forward search algorithm. In case of backward search algorithm, a feature type will be discarded even if it only offers a slight change in performance to keep the complexity lower. The forward search approach is preferable as it goes from low complexity to high, whereas the backward search starts with high overhead in complexity and reduces as it removes low performing feature types.

Mathematically, the subset selection method with backward and forward search can be expressed with equations (2.36) and (2.37) respectively.

$$j = \arg \min_i MAE(F - f_i) \quad (2.36)$$

The established subset is denoted by F and f_i represents a feature type to be removed from the subset.

$$j = \arg \max_i acc(F - f_i) \quad (2.37)$$

Remove f_i if $acc(F - f_i) \geq acc(F)$, for complexity reason it includes the equal sign and those feature types which cannot change the performance significantly.

Although the order of complexity is the same between the two search algorithms, forward search algorithm is more preferable as it emerges from simple to complex. And it is mathematically described as the addition of feature types one by one in a greedy manner in equation (2.38).

$$j = \arg \min_i MAE(F + f_i) \quad (2.38)$$

For classification we use equation (2.39) as shown below.

$$j = \arg \max_i acc(F + f_i) \quad (2.39)$$

Add f_i to the subset F if $acc(F + f_i) > acc(F)$

In order to make the process more visible to readers, we took a simple example briefly explained in Ethem Alpaydin's machine learning book [85]. Using the nearest mean [88] as a classifier on the Iris dataset [89] with single feature, the accuracies were 76%, 57%, 92%, and 94%, for sepal length, sepal width, petal length and petal width respectively. We pick the single feature which showed the highest accuracy i.e. petal width and add one more feature to see their combined effect. Assuming $\{F_1, F_2, F_3, \text{ and } F_4, \}$ represent the four features, Figure 2.9 below shows the accuracies of the classification using two features combined.

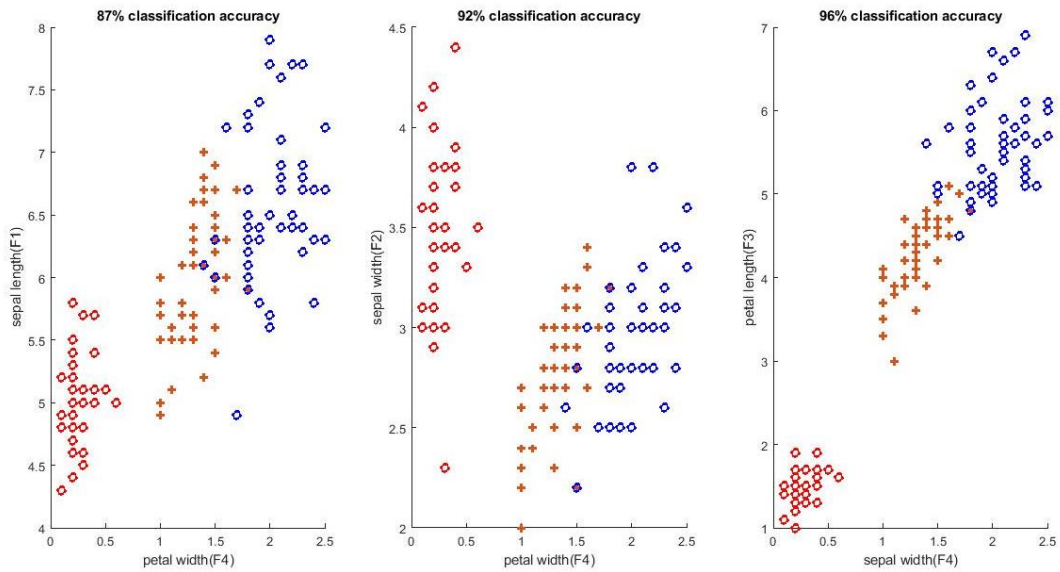


Figure 2.9. Classification of Iris data with two features using nearest mean

As we can observe from the graphs in Figure 2.9 above, combining F_3 and F_4 has improved the accuracy by 2% compared to the best single alone feature. This process however, takes considerably large amount of time and processing complexity for multi feature and large data. It takes $d + d - 1 + d - 2 + d - 3 + \dots + d - k$ training and testing sessions which makes the order of the complexity $o(d^2)$ to reduce the dimension d to k and obtain better classification accuracy.

2.5.2. Principal component analysis (PCA)

With minimum loss of information data can be represented in $d - k$ fewer dimensions than in its original d dimensions [90]. This process is named as principal component analysis (PCA) and it projects an original data X in to the direction of W to generate new feature sets Z with new dimension k , where $k < d$. PCA does not depend on output information to maximize the variance between observations as defined by equation (2.40).

$$Z = W^T X \quad (2.40)$$

The aim of PCA is to maximize variance and create reparability between data and eventually make the difference between sample points become apparent. For a unique solution the magnitude of the principal component W need to be unity i.e. $\|W_1\| = 1$. If we want to maximize the separability we need to maximize the variance $Var(Z_1)$ of the newly transformed matrix $VarZ_1$ using Lagrange multipliers problem solving techniques using equation (2.41) below [91].

$$Var(Z_1) = W_1^T \Sigma W_1 \quad (2.41)$$

With the constraint $W_1^T W_1 = 1$ we maximize the variance with the following approach defined in equation (2.42):

$$\max_{W_1} \{W_1^T \Sigma W_1 - \alpha(W_1^T W_1 - 1)\} \quad (2.42)$$

The partial derivative w.r.t. W_1 , leads us to visualize the local maxima when the slope is zero as described in equation (2.43)

$$\frac{\partial \{W_1^T \Sigma W_1 - \alpha(W_1^T W_1 - 1)\}}{\partial W_1} = 0 = 2\Sigma W_1 - 2\alpha W_1 \quad (2.43)$$

Therefore, $\Sigma W_1 = \alpha W_1$, which clearly shows that W_1 is an Eigen vector and α is a corresponding Eigen value [92] that can maximize the argument in equation (2.42). Hence, $W_1^T \Sigma W_1 = \alpha W_1^T W_1 = \alpha$.

The transformation matrix W is formed from the k Eigen vectors $\{W_1, W_2, W_3, W_4, W_5 \dots \dots, W_k\}$ concatenated based on the order of their corresponding Eigen value $\{\lambda_1, \lambda_2, \lambda_3, \lambda_4, \lambda_5 \dots, \lambda_k\}$ where $\lambda = \alpha$.

2.5.3. Linear discriminant analysis (LDA)

Unlike PCA linear discriminant analysis (LDA) involves a supervised dataset to reduce the dimension of feature sets used to represent each observation of a data sample from d to $N - 1$, where N is the number of classes. LDA begins with a two class problem and generalizes it for multiple classes more than two. Therefore, the dimension reduction is from d to 1.

Again vector W transforms observations X drawn from two classes C_1 and C_2 to reduce their dimension to 1. The transformation is on to the direction of W where the reduced observations will be $Z = W^T X$. Given the sample observations $X = \{x^t, r^t\}$, such that $r^t = 1$ if x^t is drawn from class 1 and $r^t = 0$ if x^t is drawn from class 2.

Let us assume M_1 and m_1 represent means of our data drawn from class 1 before and after dimensionality reduction respectively. Like wise M_2 and m_2 represent means of class 2 C_2 and given by equation (2.44) below.

$$m_1 = \frac{\sum_t W^T x^t r^t}{\sum_t r^t} = W^T M_1 \quad (2.44)$$

The mean after transformation for class 2 is

$$m_2 = \frac{\sum_t W^T x^t (1-r^t)}{\sum_t (1-r^t)} = W^T M_2 \quad (2.45)$$

After projection the scatter of samples S_1^2 and S_2^2 from class C_1 and C_2 respectively are given by:

$$S_1^2 = \sum_t (W^T x^t - m_1)^2 r^t \quad (2.46)$$

The scatter in class 2 is:

$$S_2^2 = \sum_t (W^T x^t - m_2)^2 (1 - r^t) \quad (2.47)$$

In order to make the two classes well separated, we need to maximize the difference between the means of the two classes after projection. To make the two classes easily separable the means should be as far apart as possible, but scattered in as small region as possible. Using Fisher's linear discriminant, the function to achieve both requirements can be stated as in equation (2.48) below:

$$J(W) = \frac{(m_1 - m_2)^2}{S_1^2 + S_2^2} \quad (2.48)$$

When we explicitly express the numerators and denominators of the above equation in terms of original data means and other parameters;

$$\begin{aligned} (m_1 - m_2)^2 &= (W^T M_1 - W^T M_2)^2 \\ &= W^T (M_1 - M_2)(M_1 - M_2)^T W \\ &= W^T S_B W \end{aligned}$$

Where $S_B = (M_1 - M_2)(M_1 - M_2)^T$ represents between-class scatter matrix [93]. When we look at the denominator in equation (2.48), the scatters can be expressed with original data as:

$$\begin{aligned} S_1^2 &= \sum_t (W^T x^t - m_1)^2 r^t \\ &= \sum_t W^T (x^t - M_1)(x^t - M_1)^T W r^t \\ &= W^T \{ \sum_t (x^t - M_1)(x^t - M_1)^T r^t \} W \\ &= W^T S_1 W \end{aligned}$$

Where $S_1 = \sum_t (x^t - M_1)(x^t - M_1)^T r^t$ represents within-class scatter matrix for class 1. Similarly $S_2 = \sum_t (x^t - M_2)(x^t - M_2)^T r^t$ expresses the scatter matrix for class 2. Finally the total within-class scatter matrix S_w is the sum of the scatters in the two classes $S_1 + S_2$ which can be expressed as in equation (2.49):

$$\begin{aligned} S_1^2 + S_2^2 &= W^T S_1 W + W^T S_2 W \\ &= W^T (S_1 + S_2) W \end{aligned} \quad (2.49)$$

To compute the transformation matrix W , we need to maximize the function $J(W)$. Hence after taking the partial derivative of $J(W)$ w.r.t W we set it equal to zero.

$$\frac{\partial J(W)}{\partial W} = 0 = 2 \frac{W^T (M_1 - M_2)}{W^T S_W W} \left\{ (M_1 - M_2) - \frac{W^T (M_1 - M_2)}{W^T S_W W} S_W W \right\} \quad (2.50)$$

Given that, the mathematical expression $\frac{W^T (M_1 - M_2)}{W^T S_W W}$ is a constant, we have:

$$W = C S_W^{-1} (M_1 - M_2) \quad (2.51)$$

The constant C shows the magnitude. However, we are more interested in the direction instead. Hence the constant can be assumed to be 1 and the transformation matrix will be well computed as $W = S_W^{-1}(M_1 - M_2)$ with $C=1$. For normal distribution $p(x|C_i) \sim N(\mu_i, \Sigma)$, the transformation matrix is expressed as $W = \Sigma^{-1}(\mu_1 - \mu_2)$.

For multiple classes $k > 2$, the linear transformation can be generalized by redefining the between-class and within-class scatter matrices. The dimensionality reduction is from d to k removing $d - k$ less relevant dimensions. The transformation matrix W becomes $d \times k$ instead of $d \times 1$.

We compute the scatter matrix S_i for each class C_i , where $i = 1, 2, 3 \dots, k$ and k the number of classes using equation (2.52) below.

$$S_i = \sum_t r_i^t (x^t - M_i)(x^t - M_i)^T \quad (2.52)$$

The class label $r_i^t = 1$ if the observation x^t is drawn from class C_i otherwise $r_i^t = 0$. Hence the total within-class scatter is the superposition of all the individual within-class scatter matrices in each class as defined by equation (2.53).

$$S_W = \sum_{i=1}^k S_i \quad (2.53)$$

Between-class scatter matrix S_B determines how far apart are the means of each class from the overall mean of the data.

$$S_B = \sum_{i=1}^k N_{C_i} (M_i - M)(M_i - M)^T \quad (2.54)$$

The term $N_{C_i} = \sum_t r_i^t$ denotes the total number of observations in each class C_i in equation (2.54) above. The $k \times k$ matrices $W^T S_B W$ and $W^T S_W W$ represent between-class and within-class scatter matrices after projection respectively. Again we need to formulate a function that maximizes the between-class and minimizes within-class scatter simultaneously in order to make the data easily separable between classes. If we put both matrices in a rational function at the numerator and denominator as shown in equation (2.55) respectively, then maximize this function can successfully achieve the two objectives at the same time.

$$\max_W J(W) = \max_W \left\{ \frac{W^T S_B W}{W^T S_W W} \right\} \quad (2.55)$$

Taking the partial derivative of the function $J(W)$ and set it equal to 0 leads us to find the optimal solution W . The largest eigenvectors computed from the product $S_W^{-1}S_B$ provide the optimal solution to both problems stated above simultaneously.

2.5.4. Feature fusion

We used simple concatenation of the ten feature sets in an intertwined manner. In fact the result was not better compared to a single feature best performance. Therefore, we reduced the least performing feature sets until we finally obtain encouraging performances from our classifiers in a subset backward search feature selection manner. Accordingly, we obtained the best performance after removing the three least performing feature types.

The feature types eventually dropped from the fusion include IMFCC, cosine phase and SSFC which are described in this chapter. The fusion combined three classes of feature types; filter bank based, sub-channel based and spectral phase-based features to form a vector with higher dimension for each frame.

We used different form of concatenation to combine elements of each feature set. Iterative algorithms instead of the traditional vertical or horizontal concatenation are implemented to arrange each element of feature sets in an intertwined manner one after the other. The pattern of 7 feature sets obtained in such a manner has improved the accuracy of speaker age classification using the cosine distance scoring (CDS).

3. EMBEDDING WITH MACHINE LEARNING AND DEEP LEARNING MODELS

Acoustic word embedding (AWE) in speech processing applications is a fixed-dimensional representation of variable-length utterances in an embedding space [94]. Therefore embedding is a relatively low-dimensional space into which high-dimensional vectors can be translated. Machine learning classification, regression or any approximation models on large inputs like sparse vectors representing words are made easier mainly due to embedding. Two embedding schemes namely; i-vector and x-vector are employed in our research. These vectors are briefly presented in the following two sub sections consequently [55-57].

3.1. i-Vector Embedding

A vector of fixed dimensions from variable length utterances is generated via four major steps depicted in Figure 3.1 below.

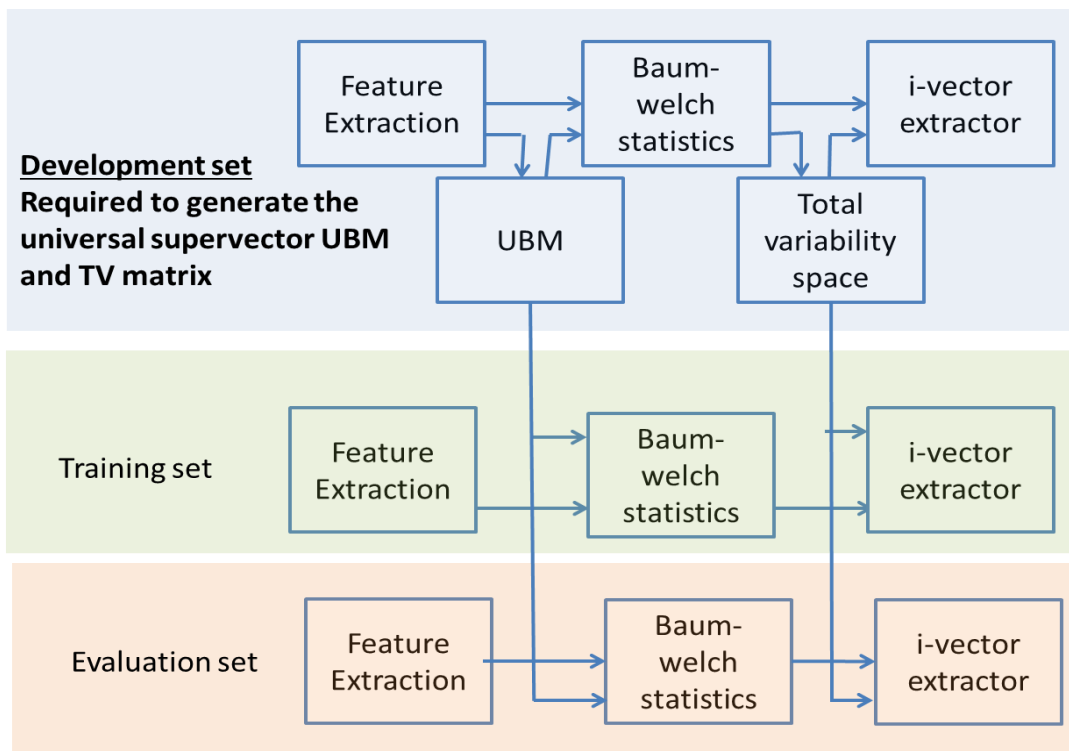


Figure 3.1. i-Vector extraction

Any of the feature sets presented in Unit 3 can be used in the generation of these vectors. The i-vectors are defined alongside with a factor analysis which uses a linear model to compute feature matrices. They are basically sets of vectors generated based on factor analysis in which the acoustic features (typically MFCC and log-energy plus their 1st and 2nd derivatives) by a Gaussian mixture model (GMM). Suppervector (UBM) M and total variability (TV) T are determined from the development set using expectation maximization algorithm [95]. Therefore each utterance X_i from a dataset $X = \{X_1, X_2, X_3, \dots, X_N\}$ is written in terms of a global mean M , a low-rank $R \times D$ matrix T , D -dimensional latent factor ω_i which eventually represents i-vectors with prior density $N(\omega|0, I)$ and residual noise ϵ_i following a Gaussian density with zero mean and covariance matrix Σ .

$$X_i = M + T\omega_i + \epsilon_i \quad (3.1)$$

The marginal distribution of X is given by.

$$p(X) = \int p(X|\omega)p(\omega)d\omega \quad (3.2)$$

Substituting the normal distributions in the above equation:

$$p(X) = \int N(X|M + T\omega, \Sigma)N(\omega|0, I)d\omega \quad (3.3)$$

Assuming the above equation as the convolution of Gaussians it finally leads to the following mathematical expression.

$$p(X) = N(X|M, TT^t + \Sigma) \quad (3.4)$$

Determining the parameters M , T , and Σ needs an iterative method called expectation maximization which consists of two steps: the expectation E-step computes parameters and the maximization M-step maximizes these parameters [96].

3.2. Deep Learning-Based Embedding

3.2.1. Introduction to deep learning

Deep learning is the 21st century's most exploited subset of artificial intelligence (AI). Most tech giants such as Google, Intel, Microsoft, Facebook, Apple, Twitter and others extensively apply deep learning models in their daily activities [97]. The deep

learning concept re-emerged long after Rosenblum's perceptron proposed in 1958 failed to recognize multiple classes. A perceptron is a single layer artificial neural network capable of learning linearly separable patterns. Therefore, it could not live up to its expectations. For this reason, the neural network research had stagnated until it finally showed a resurgence in the 1980s with the emergence of multilayer feed-forward neural networks which showed significant improvement in processing power over perceptron. But most kept faith in it and recently it has become a trend in the research community due to additional hidden layers introduced to it in order to make it able to learn from complex data distribution [29].

Transformation and extraction of features are usually associated with deep learning algorithms whereas neural networks use neurons to fire data in the form of input and output values via connections. Section 4.4 in the next chapter briefly discusses DNNs in detail. It compares the biological nervous system with artificial neural networks (ANNs).

The deep neural network (DNN) structure shown in Figure 3.2 below depicts how each of the input components, the neurons in hidden layers and outputs are connected to each other. DNN algorithms basically filter out attributes associated with labels or actual values in what exactly resembles as a data distillation process. The mathematical details will be presented in-depth in the next chapter but in the meantime the number of connections between two adjacent layers is a product of a number of neurons (the circular structures in the figure) in the two layers.

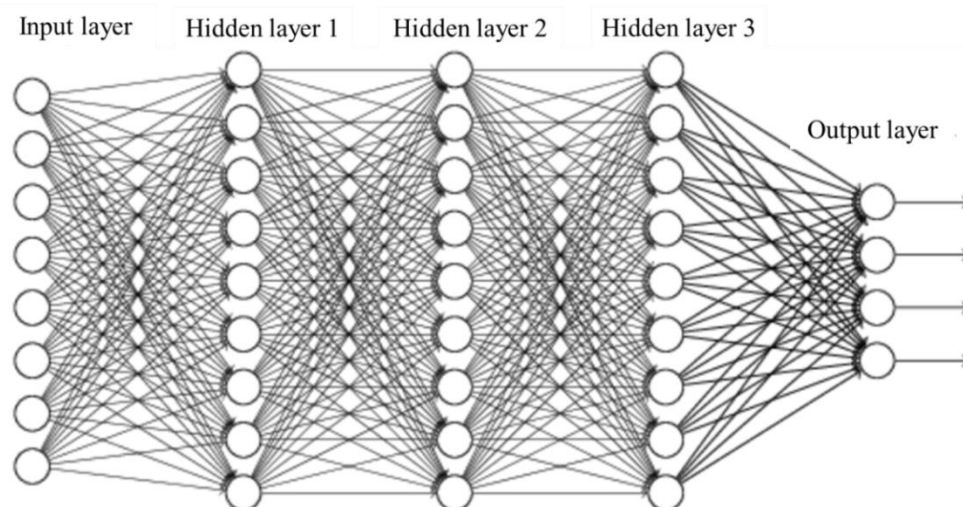


Figure 3.2. Multilayer deep neural network structure

The weights associated with each connection or synapses as in the biological nervous system are stored in an $N \times M$ matrix where M stands for the number of neurons in the layer found at the left and N represents the number of hidden units (neurons) in the layer closer to the output at right.

Deep learning algorithms apply the Keras toolkit to classify supervised data. The Keras deep learning framework is a model-level library, providing high-level building blocks for developing deep-learning models. It runs on top of Tensorflow, Theano or Microsoft Cognitive Toolkit (CNTK) backend.

3.2.2. x-Vector embedding

The number of frames in each utterance makes a random variable that might not be expressed using known distributions. The x-vector architecture converts these variable length feature sequences in utterances into a fixed-dimension embedding which contains the relevant information of the utterance. The x-vector embedding is extracted by a temporal pooling layer in time delay neural networks (TDNNs) which summarizes information along the time axis [98]. This kind of network is starting to outperform the state-of-the-art i-vector embedding in tasks like speaker and language recognition. Its reputation is believed to be mainly due to context level processing. After getting this embedding, utterance level labels, such as speaker identity, age, and gender, can be used for discriminative network training. Thus, end-to-end training becomes possible, jointly optimizing both feature extraction and prediction [62].

During the processing of a wider temporal context, in a standard DNN, the initial layer learns an affine transform for the entire temporal context. Affine transform is a geometric transformation which preserves lines and parallelism but not necessarily distance and angles. However in TDNN architecture the initial transforms are learned on narrow contexts and the deeper layers process the hidden activations from a wider temporal context. Hence the higher layers can learn wider temporal relationships. Each layer in a TDNN operates at a different temporal resolution, which increases as we go to higher layers of the network. Each of the neurons in the subsequent layers learns from a sampled set of neurons in the previous layer. Contexts increase as the

process goes deeper in to higher layers. This shows the neurons in the higher layers get a wider context than those in lower layers as is shown in Figure 3.3 below [98].

The transforms in the TDNN architecture are tied across time steps and for this reason they are seen as a precursor to the convolutional neural networks. During back-propagation, due to tying, the lower layers of the network are updated by a gradient accumulated over all the time steps of the input temporal context. Thus the lower layers of the network are forced to learn translation invariant feature transforms [99].

The input contexts of each layer required to compute output activation, at one time step define the hyper-parameters which describe the TDNN network. x-Vector is a greedy approach that can only perform better than i-vector with large amount of data. Hence we incorporated additional data from MUSAN database consisting of music, speech and noise [100] as well as the simulated room impulse response (RIRs) data base for augmentation [101].

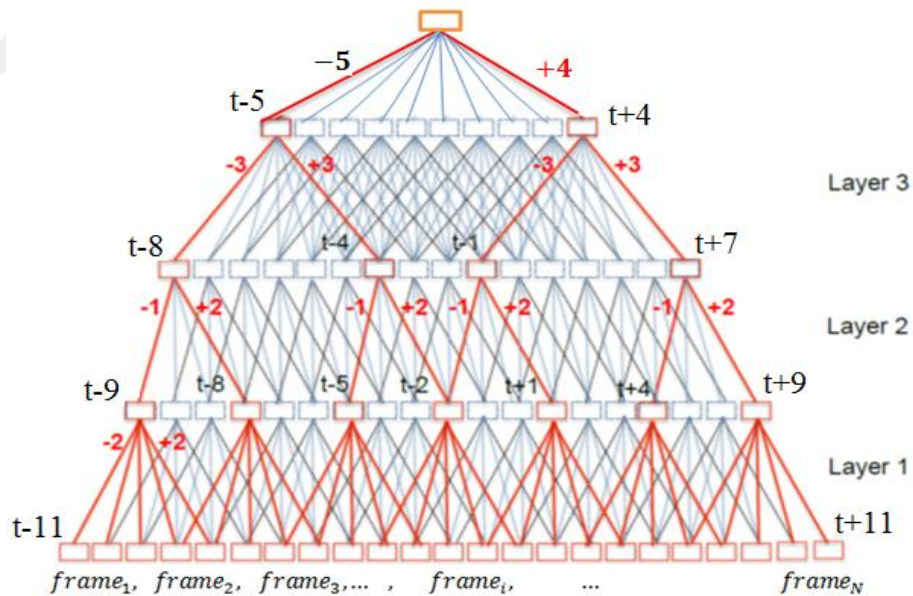


Figure 3.3. Time delay neural net (TDNN) Computation with sub-sampling (red) and without sub-sampling (blue+red)

The above figure shows the time steps at which activations are computed, at each layer, and dependencies between activations across layers. It can be seen that the dependencies across layers are localized in time. In addition table 1 below describes layer-wise context specification, corresponding to the TDNN shown in fig. 4 above.

Hence the first neuron in the second layer connects with two neurons forward and two neurons back ward in the first layer forming a symmetrical triangle. Therefore the context will be $[-2,2]$ at the first layer for instance. If the forward meaning future frame index and back ward previous frame indices are not equal the triangle formed will not be symmetrical rather its apex moves towards the lower index in absolute value.

Once the frame level computation is done the statistics pooling layer continues and all the proceeding layers until the softmax layer including the statistics layer process on segment level data [56]. Table 3.1 below shows the context size in each layer and the total context size from the apex.

Table 3.1. Context specification of the TDNN shown in fig. 4 above

Layer	Input context(the red lines) at every layer	Total context size
1	$[-2, +2] = \{t-2, t, t+2\}$	5
2	$[-1, +2] = \{t-1, t, t+2\}$	$8 = \{t-3, t, t+4\}$
3	$[-3, +3] = \{t-3, t, t+3\}$	$14 = \{t-6, t, t+7\}$
4	$[-5, +4] = \{t-5, t, t+4\}$	$23 = \{t-11, t, t+11\}$
5	$\{0\}$ this means at frame t	$23 = \{t-11, t, t+11\}$

The fixed dimensional x-vectors which uniquely represent the age characteristics of a speaker can be extracted at any layer after the statistics pooling layer but before the softmax layer. The complete end-to-end flow of the whole age estimation approach is depicted in fig. 5 below.

Algorithm 3.1 below presents the sample command creating the DNN layers which all perform TDNN. It is an excerpt taken from our Kaldi code series. Accordingly it specifies the input dimension as the dimension of the MFCC feature set which is, 42 for a frame of speech. With these terminal inputs it creates 3 frame level and 2 segment level layers. x-Vectors are pulled out at the fourth or fifth layers then fed to a softmax or other classification model.

At the input layer, we provide frames each of which consists of a set of a fixed set of features. In our experiments each frame consists of 13 MFCC, 13 dynamic, 13 acceleration and one more feature as an energy component from each of the static, dynamic and acceleration MFCC features.

Algorithm 3.1

the frame-level layers implemented using python 3.6

input dim=\${feat_dim} name=input

relu-batchnorm-layer name=tdnn1 input=Append(-2,-1,0,1,2) dim=512

relu-batchnorm-layer name=tdnn2 input=Append(-2,0,2) dim=512

relu-batchnorm-layer name=tdnn3 input=Append(-3,0,3) dim=512

relu-batchnorm-layer name=tdnn4 dim=512

relu-batchnorm-layer name=tdnn5 dim=1500

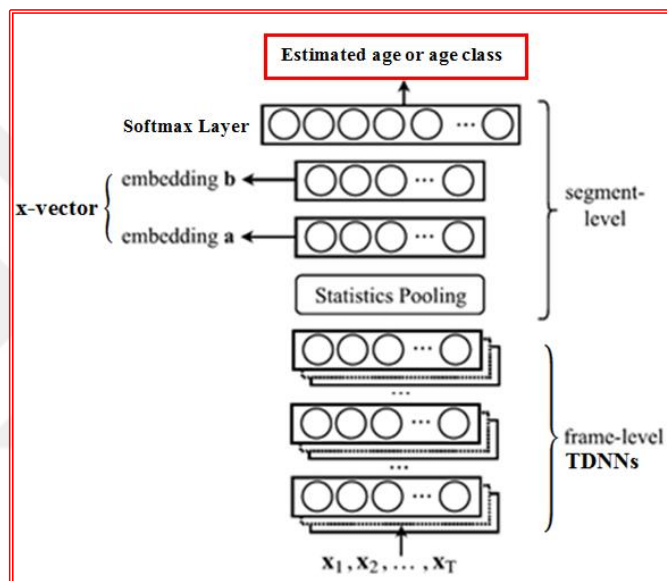


Figure 3.4. TDNNs to softmax end-to-end speaker age estimation

Hence a total of 42 features are used to represent a given frame. The DNN establishes the context and learns important traits from connecting all these subsampled clusters of frames in the frame level processing at the higher layers. The network eventually generates a set of fixed dimensional feature sets that can represent all the frames in a condensed manner. This set, of features are widely known as x-vectors in the speech processing community recently.

4. CLASSIFICATION AND REGRESSION MODELS

Due to its ease of implementation and superiority of its applications, most of our experiments give relatively better focus to speaker age classification rather than regression. However, regression studies are also carried out to some extent. Plenty of human desires come categorically which fundamentally consider age groups. Some services, ads, online contents, entertainments etc. are unpleasant to a certain age group while others could enjoy them greatly.

Classification is a decision process which employs relatedness or closeness to categorize observations in to different labels based on parameters determined from collected data. It is also a process of learning patterns [102]. For instance we can decide whether a student fails or passes his examination based on criteria collected from a wide range of experiences. These criteria are the factors affecting student's ability to pass examinations. To list a few factors: the number of hours studied, the number of hours slept, health status, the broadness of the content studied, the difficulty of the exam and others. Putting all these factors in to numbers and generating new parameters such as mean and variance or covariance we could be able to classify students' ability before the results are displayed. Similarly a business plan could be predicted whether it is risky or reasonable based on factors such as market condition, initial investment and so on [85]. The same holds for age classification in which we train our model with a training set speech dataset. Extracted features with their respective labels (ages) will be used to learn the model we develop. A brief discussion has been made on some of the famous classification models.

4.1. Gaussian Mixture Model (GMM)

The Gaussian mixture model (GMM) is a collection of weighted multivariate Gaussian distributions. This model assumes there is a certain number of clusters in unsupervised data that tend to show Gaussian distribution with distinct parameters. It is a superposition of all independent and identically distributed random variables

(i.i.d.) which satisfy Gaussian distribution with each of them having their own parameters scaled by different weights. The Gaussian distribution is the most common probability distribution in statistics. The graph of the Gaussian distribution as shown in Figure 4.1, is a bell shaped curve where the highest probability occurs when the value of the random variable is equal to the mean value (μ). The distribution is commonly called as the normal distribution. It is designed in such a way that 68% of the populations or samples are included in one standard deviation range i.e. the probability of finding the random variable x between $\mu - \sigma$ to $\mu + \sigma$ is 0.68. Similarly the probability of finding x in two standard deviation ranges which is between $\mu - 2\sigma$ to $\mu + 2\sigma$ gets higher and is 0.95. Increasing the range to three standard deviations makes the probability of finding the random variable in this range to 99.7% (0.997). This clearly shows most of the observations accumulate around the mean. The mathematical computation of the Gaussian distribution is given in equation (4.1) below.

$$p(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \quad (4.1)$$

Where x is independent and identically distributed (i.i.d.) random variable and in this case it is a continuous variable, e is a natural exponent whose value is agreed to be 2.71828, μ is the mean of the random variable and σ^2 is the variance of the random variable. The variance is the square of standard deviation (σ).

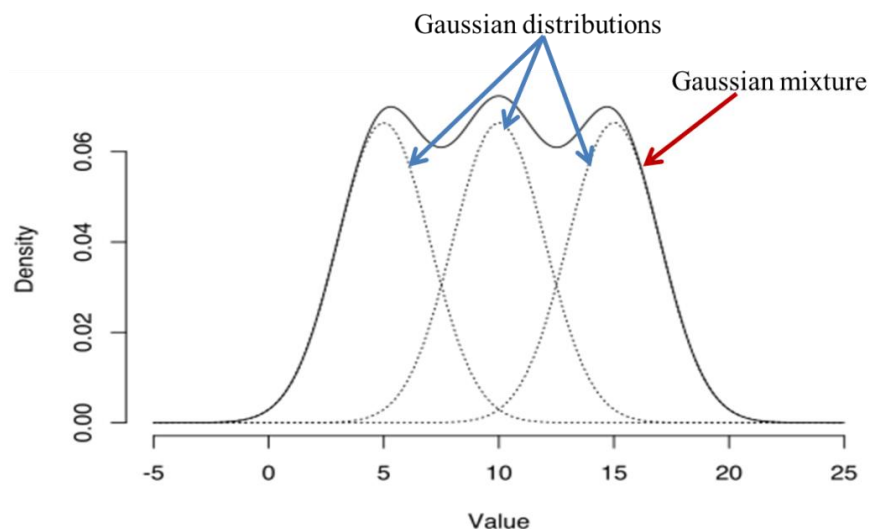


Figure 4.1. Gaussian distributions and mixture

When there are more distributions with different parameters; μ_i and σ_i where $i = 1, 2, 3, \dots, N$ here N is the number of mixtures all the Gaussian distributions should be scaled with corresponding weights based on which distributions dominate and which ones have less influence and finally superposition is applied to the distributions to provide the model called Gaussian Mixture model. The scaling factors are given by weight variables; w_i . The mathematical expression for the Gaussian mixture is given in equation (4.2) below.

$$p(x, \lambda) = \sum_{i=1}^N w_i \frac{1}{\sqrt{2\pi\sigma_i^2}} e^{-\frac{(x-\mu_i)^2}{2\sigma_i^2}} \quad (4.2)$$

For data expressed with d features, the multivariate Gaussian normal distribution is expressed as shown in equation (4.3) below.

$$N(X|\mu, \Sigma) = \frac{1}{\sqrt{(2\pi)^d |\Sigma|}} e^{-\frac{1}{2}(X-\mu)^T \Sigma^{-1} (X-\mu)} \quad (4.3)$$

The mixture of k clusters would form the probability distribution given by equation (4.4) [103]:

$$P(X|w, \mu, \Sigma) = \sum_{i=1}^k \left\{ w_i \frac{1}{\sqrt{(2\pi)^d |\Sigma_i|}} e^{-\frac{1}{2}(X-\mu_i)^T \Sigma_i^{-1} (X-\mu_i)} \right\} \quad (4.4)$$

The expectation-maximization (EM) algorithm is employed to compute the Gaussian parameters $\lambda_i = \{w_i, \mu_i, \Sigma_i\}$ for each cluster C_i [104]. This helps to develop the GMM that predicts the cluster of a certain observation using posterior probability $P(C_i|X)$ from incomplete data with missing attributes called latent variables [105]. The EM algorithm has been exploited for a variety of application areas and continuously upgraded to perform faster [106].

4.2. Cosine Distance Scoring with i-Vector (CDS)

The cosine distance scoring (CDS) is a score given to a test speech sample after determining the cosine distance between the test sample and target class [14]. The i-vectors are determined for each utterance. Then the average of i-vectors is calculated for each target class as in equation (4.5). Every test i-vector is scored against target class i-vectors as shown in equation (4.6).

$$\omega_{tarclass_i} = \frac{1}{M_i} \sum_{k=1}^{M_i} \omega_{train_k} \quad (4.5)$$

$$\text{COS_SCORE}_{k_i} = \frac{\omega_{test_k}^T \omega_{tarclass_i}}{\|\omega_{test_k}\| \|\omega_{tarclass_i}\|} \quad (4.6)$$

In the above equations, ω_{train_k} is i-vector for training sample k in the i^{th} target class, i is the specific target class, $= \{1, 2, 3, \dots, N\}$, $\omega_{tarclass_i}$ is the average i-vector for target class i and N is the number of target classes in equations (4.3) and (4.4). And M_i is the number of training samples in target class i . In addition, cos_score_{k_i} stands for cosine distance scoring between test sample k and target class i .

4.3. Probabilistic Linear Discriminant Analysis with i-Vector (PLDA)

Probabilistic linear discriminant analysis (Probabilistic LDA, a.k.a. PLDA [107]) is a generative approach that tries to create data instances for a given class with Gaussian distributions [108]. LDA is deterministic and models intra-class and inter-class variations as multidimensional Gaussians while PLDA is a probabilistic approach and assumes data instances come from Gaussian distributions [109]. The relationship between PLDA and LDA is analogous to that of factor analysis (FA) and principal component analysis (PCA). While the former ones are supervised PLDA being superior to LDA in modeling data instances coming from unseen classes, the later are unsupervised.

When we say PLDA is generative it means it captures or learns the joint probability distribution $p(x_1, x_2)$ of observations assumed as data instances from a mixture of distributions without labels. Unlike PLDA, discriminative models capture the conditional probability of outcomes given data instances $p(y|x)$ commonly known as posterior probability. The distribution of the latent variables y usually invisible but most powerful in representing outcomes of a given model for a certain class can be generated using the famous Gaussian distribution with mean μ and semi-definite between-class covariance Σ_b .

We use a non-singular transformation matrix V to convert the between-class Σ_b and the definite within-class Σ_w covariance matrices in to diagonal matrices in order to transform data instances from their original feature space to a dimensionally reduced latent space. As well discussed in section 2.4 above, the objective of this

transformation is to reduce dimensionality and to create separability between classes. The advantage of PLDA over LDA is that it allows making inferences about classes that are unseen during training sessions [109]. The two covariance matrices apparently expressing definite scatters within classes Σ_w and semi-definite scatters between classes Σ_b can be diagonalized with generalized eigen problem simultaneously as:

$$\begin{aligned} V^T \Sigma_w V &= I \\ V^T \Sigma_b V &= \psi \end{aligned} \tag{4.7}$$

where I and ψ are identity and diagonal matrices that can be computed with eigen matrix decomposition in equation (4.7) above.

Observed variables x and y in the feature space can be expressed with latent variables u and v in the latent space which is the transformed domain with normal distributions, $u \sim N(\cdot | v, I)$ and $v \sim N(\cdot | 0, \psi)$ respectively as:

$$\begin{aligned} x &= \mu + Au \\ y &= \mu + Av \end{aligned} \tag{4.8}$$

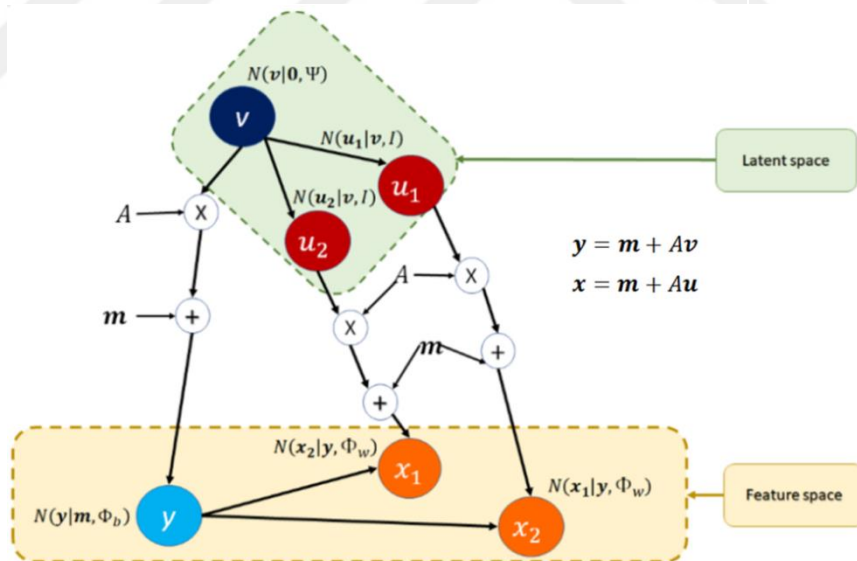


Figure 4.2. Projection of observed features in to latent space

where, the matrix A can be computed from the non-singular matrix V as $A=V^{-T}$ and the latent variables v and u represent class and data instances of the class in the projected domain in equation (4.8) shown above. This is demonstrated in Figure 4.2 above. The figure is taken from a website named as “towards data science”.

The i-Vectors are used as observed variables both in the CDS and PLDA classifiers. They are identity vectors extracted from a joint factor analysis (JFA) expression of an utterance [110]. A super-vector M consisting of speaker and channel or session subspaces. The speaker dependent super-vector is defined as shown in (4.9).

$$M = m + Vy + Ux + Dz \quad (4.9)$$

In equation (4.7), m denotes session-independent speaker super-vector generally obtained from UBM, V and D represent eigen voice matrix and diagonal residue of speaker subspace respectively, and U denotes session subspace (Eigen channel matrix). The vectors x, y, z are assumed to be random variables with a normal distribution $N(0, I)$. They are speaker, channel and residual factors in their respective subspaces [111]. A new space referred to as “total variability space,” that contains speaker and channel variability simultaneously is proposed in [110]. It is defined by the total variability matrix that contains the eigenvectors with the largest eigenvalues of the total variability covariance matrix[112]. Accordingly for a given utterance the new space redefines the GMM super-vector M as:

$$M = m + T\omega \quad (4.10)$$

where m is the new speaker channel-independent universal super-vector, T is the rectangular total variability matrix of low rank and ω is an identity vector commonly known as i-vector in equation (4.10) above.

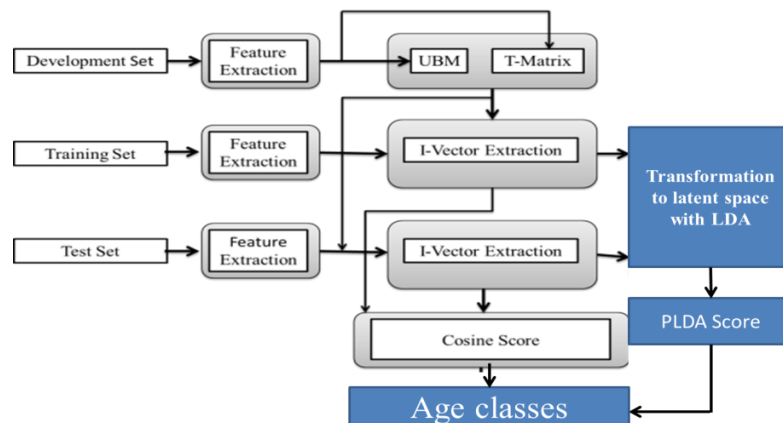


Figure 4.3. Overall process diagram of CDS and PLDA Classifiers

The overall block diagram of CDS and PLDA classifiers is shown in Figure 4.3 below. The development, training and test sets are all common to both classifiers. Procedures applied to the test set are identical until the scoring execution.

The development set is used to generate the total variability space matrix and the UBM super-vector which is common in all the three classifiers is used. The training set undergoes a similar process until i-vector extraction phase. After this phase it is directly used by the cosine score whereas it will be transformed using LDA in the PLDA classifier.

4.4. Deep Neural Network (DNN) Based Classifiers

Neural networks formally known as artificial neural networks (ANN), were initially proposed by psychologist Frank Rosenblatt in 1958 to imitate the processing power of the human brain. They were called perceptrons and mainly used to process visual data and learn to recognize objects. They were not successful at the beginning and therefore most scholars were not patient to wait while very few kept faith in their performance. Eventually neural networks began to outperform what can be done with traditional machine learning models.

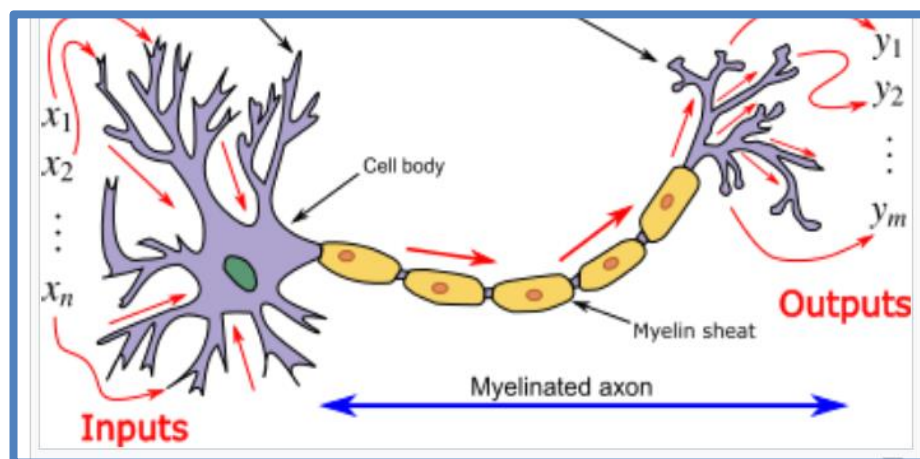


Figure 4.4. Biological neural network

The nodes in modern day neural nets represent biological neurons in the nervous system mostly located in our brain. The connection lines (edges) connecting nodes are analogous to the synapses in the biological brain as depicted in Figure 4.4 taken from Wikipedia. These connections transmit signals from it input to neurons or from one neuron to another where all the signals arriving at a neuron are summed up and

fed to a non-linear activation function. All the connections are associated with randomly picked initial weights which get updated as training proceeds.

Activation functions determine the outcome of a given neuro based on a set of inputs and weights associated with the connections to the neuron. The most popular activation function is the sigmoid activation function also known as logistic activation function in its other name. Logistic or sigmoid or else soft step activation function is defined as in equation (4.11) below.

$$\varphi(\zeta) = \frac{1}{1 + e^{-\zeta}} \quad (4.11)$$

The Greek symbol ζ represents the sum of all the inputs to a certain neuron multiplied by their respective weights and a bias parameter b on top of it as computed in equation (4.12) below. The bias is not associated with any input variable or intermediate neuron output.

$$\zeta = b + \sum_{i=1}^{N_c} x_i \omega_i \quad (4.12)$$

The parameters N_c and ω_i represent the number of synapses connected to the neuron and the weight of the i^{th} connection respectively while x_i denotes input feature or an output from a node in the previous layer associated to the connection. Figure 4.5 below presents the entire network consisting two layers with five hidden units or neurons each.

Let us assume a simple 4 input and 2 output single layer neural network. The input dimension is 4 and consists of two neurons at the output layer. Therefore, the input is represented as $X = [x_1 \ x_2 \ x_3 \ x_4]^T$ and the weights associated with the first and second output neuros are

$W_1 = [w_{11} \ w_{21} \ w_{31} \ w_{41}]^T$ and $W_2 = [w_{12} \ w_{22} \ w_{32} \ w_{42}]^T$ respectively where the total weight matrix can be expressed as $W^T = [W_1 \ W_2]$. Then the output value $Y = [y_1 \ y_2]^T$ of each neuron can be computed by applying an activation function to the product WX .

$$W = \begin{bmatrix} w_{11} & w_{21} & w_{31} & w_{41} \\ w_{12} & w_{22} & w_{32} & w_{42} \end{bmatrix}$$

For a generalized artificial neural network with N inputs and M activation functions, the weight matrix associated with each input to activation function pairs is given by:

$$W = \begin{bmatrix} W_{11} & W_{21} & W_{31} & W_{41} \dots W_{N1} \\ W_{12} & W_{22} & W_{32} & W_{42} \dots W_{N2} \\ W_{13} & W_{23} & W_{33} & W_{43} \dots W_{N3} \\ W_{14} & W_{24} & W_{34} & W_{44} \dots W_{N4} \\ \vdots & \vdots & \vdots & \vdots \\ W_{1M} & W_{2M} & W_{3M} & W_{4M} \dots W_{NM} \end{bmatrix}$$

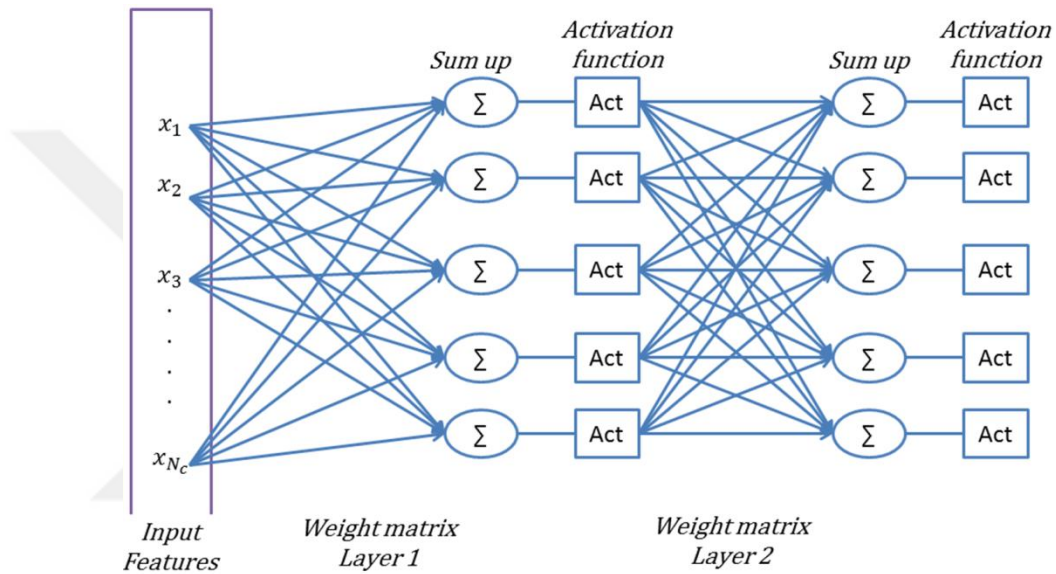


Figure 4.5. Deep neural network sample with two layers

The next major step in DNN algorithms is computing the error incurred for every training entry as a difference between predicted (estimated) and actual values. After obtaining the error values the weights will be updated to compensate for the error incurred in a backpropagation process. If we assume the output of the neurons after the first layer as $f(\cdot)$ and the second layer as $g(\cdot)$ then the eventual output is computed using the chain rule $g(f(x))$. Given the input features $X = [x_1, x_2, x_3, x_{Nc}]$ and the weight matrices in the two layers as W and W' respectively the predicted values are $g(W'f(WX))$. The weight matrices follow the size of the input features and number of hidden units where the number of columns are equal to the size of input features or the number of the hidden units to the left of the synapses whereas the rows are equal to the number of hidden units in the layer pointed by the connections to the right. The sigmoid activation function is given by equation (4.13). Its range spans from zero to one as the independent variable goes from $-\infty$ to $+\infty$.

$$f(x) = \frac{1}{1 + e^{-wx^t}} \quad (4.13)$$

The output at the second layer is defined as shown in equation (4.14) below.

$$g(f(x)) = \frac{1}{1 + e^{-w'(f(x))^t}} \quad (4.14)$$

Once predictions are made by the network, the next crucial step is to compute the error incurred by subtracting the prediction from actual values as given by equation (4.15) below.

$$e = Y(n) - \tilde{Y}(n) \quad (4.15)$$

The loss function $L(w)$ is determined as the square of the error in equation (4.15) above. It is a function of the input data, the weight and the bias parameters and defined as in equation (4.16) below. As we do not have much control over the data, we can in fact manipulate the weight and bias parameters to minimize the loss in order to obtain a prediction close to the actual value. Obtaining suitable weight and bias parameters would reduce the error ideally to zero. However, this is not possible to do it trivially using brute force optimization process as it would definitely take millions if not billions of years for an average processor [113]. In addition, the curse of dimensionality would make it unimaginable.

$$L(w) = e^2 = (Y(n) - \tilde{Y}(n))^2 \quad (4.16)$$

The optimality criteria are solving the partial derivative of the cost function $L(w)$ with respect to the weights w and obtain the weights that make the derivative zero (local minimum).

$$\frac{\partial L(w)}{\partial w} = \nabla L(w) = 0 \quad (4.17)$$

The gradient operator ∇ represents partial derivatives with respect to each weight as given in equation (4.18) below.

$$\nabla = \left[\frac{\partial}{\partial w_1}, \frac{\partial}{\partial w_2}, \frac{\partial}{\partial w_3}, \frac{\partial}{\partial w_4}, \frac{\partial}{\partial w_5}, \dots, \dots, \frac{\partial}{\partial w_m} \right]^t \quad (4.18)$$

The updated weights are determined depending on the direction of the gradient whether it is decreasing or increasing which can be visible on the sign of the gradient value. Positive and negative values of the gradient show increasing and decreasing

slopes respectively [114]. Therefore, the new weight parameters $w(n+1)$ are computed as:

$$w(n+1) = w(n) + \eta \nabla L(w) \quad (4.19)$$

where, η is a positive constant best known as the learning rate parameter or sometimes as step-size parameter in equation (4.19) above. This way all the weights get updated from output to input layers.

4.4.1. x-Vector deep neural network architecture for classification

Figure 4.6 below describes the complete process from audio inputs to accuracy computation carried out with the x-vector neural network architecture.

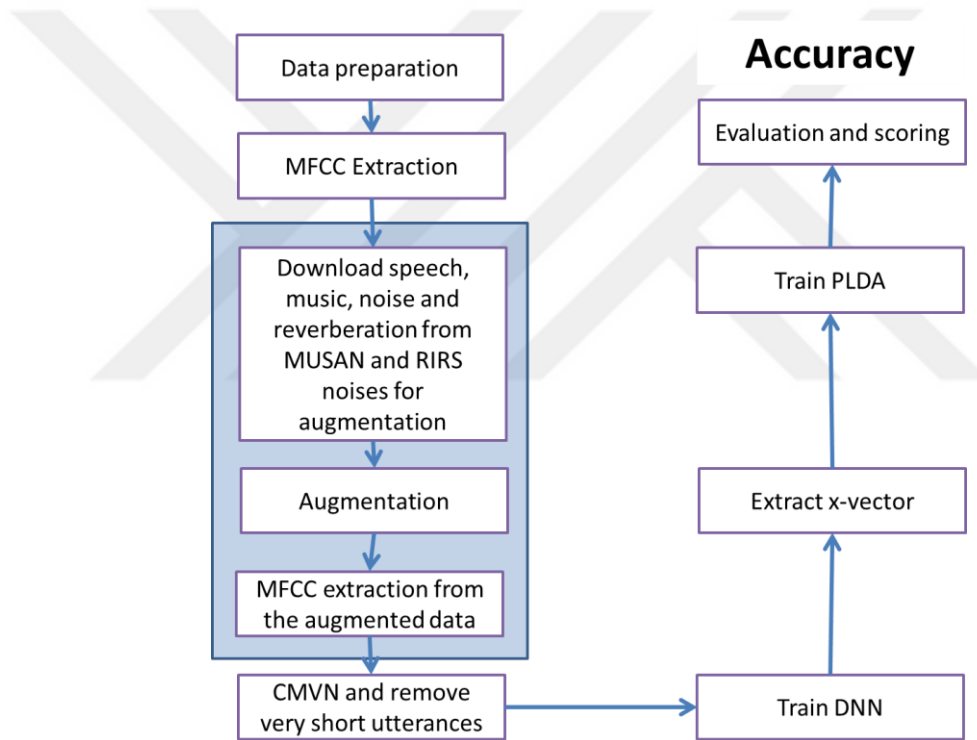


Figure 4.6. General block diagram for x-vector architecture embedding

The acoustic features MFCC and the DNN based embedding x-vector, necessary and intermediate inputs to speaker age classification as shown in Figure 4.6 above are briefly presented in Unit 2 and 3 respectively. We used the Kaldi speech recognition toolkit to implement the setup [115].

4.4.2. Long short-term memory (LSTM) networks for classification

LSTM neural network is a type of recurrent neural network (RNN) that can learn long-term dependencies between time steps of sequence data. The main components of LSTM are a sequence input layer and LSTM layer. Sequence input layer feeds sequence or time-series data into the network. This network was introduced to the machine learning world by Hochreiter and Schmidhuber in 1997. It is called short-term memory because it preserves the error that can be back-propagated through time and layers.

Figure 4.7 shown below illustrates the architecture of a simple LSTM network either for classification or regression setups. The network starts with a sequence input layer followed by an LSTM layer. To predict class labels, the network ends with a fully connected layer; a softmax layer, and a classification output layer. The regression outputs are taken at the fully connected layer. Hence the softmax layer is unnecessary for regression.

There does not exist feedback connection in Standard feedforward DNNs [116], on the other hand, LSTM has feedback connections. It can not only process single data points (such as images), but also entire sequences of data (such as speech or video).

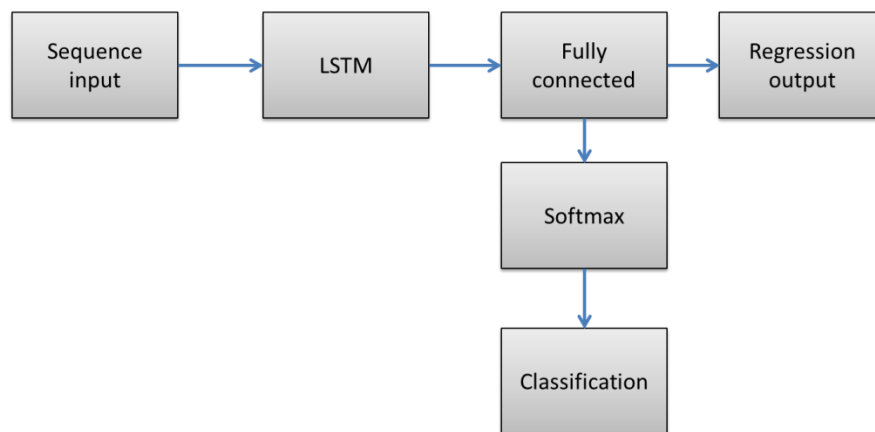


Figure 4.7. Classification and regression with LSTM

For example, LSTM is applicable to tasks such as unsegmented, connected handwriting recognition, speech recognition and anomaly detection in network traffic or intrusion detection systems (IDSs). A cell, an input gate, an output gate and a forget gate are required to compose a common LSTM. The cell remembers values

over arbitrary time intervals and the three gates regulate the flow of information into and out of the cell.

4.5. Regression Models

Regression is the process of estimating the dependent variable from a test dependent sample based on parameters obtained during the training phase. For a good regression we need a good model and a large number of supervised data. The dependent variables are often called as outcomes whereas the independent variables are called as features [13].

Alternative names for independent variables include predictors and covariates. These variables can be one or more than one and based on their number the regression is called either single variate regression or multivariate regression respectively. Linear regression is the most common and widely used kind of regression.

4.5.1. Linear regression

As it can be well depicted in Figure 4.8 below, linear regression is an effort of determining the equation of a straight line that passes through the data points. Possibly an infinite number of straight lines can pass through the data points but only one of these can show the lowest mean error. The mean error is the sum of all the distances between the true data point and the straight line or hyperplane in the case of multiple feature types whose equation is determined through the training process. The straight line becomes a hyperplane when we have more than one independent variable (feature types).

For a single variated regression analysis the two parameters determined from a training database are the slope and the vertical intercept or y-intercept in basic mathematical terms. The process of determining these parameters mainly considers plans on how to minimize the error. The ideal case is making the error down to zero.

The mathematical model is setup as shown in equation (4.20) below. Once the estimation equation is computed the error is calculated by subtracting these estimated values from each actual data point. Since we are interested in minimizing the error we deal with strategies on how to find out those parameters which can minimize the

error. In the following equations all the actual values are represented by symbols without bar and those symbols with bars are used to represent estimated values.

$$\hat{Y}_i = AX_i + C = f(X_i) \quad (4.20)$$

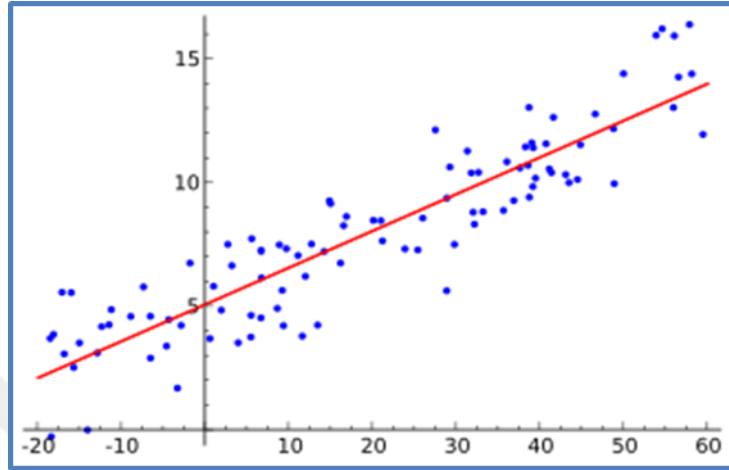


Figure 4.8. Linear regression (Picture credit to Wikipedia)

In fact the actual value of the outcome variable is a little more or a little less than the value obtained through mathematical computation using a straight line equation as it incurs an irreducible error ϵ_i for each data point computation (X_i, Y_i) .

$$Y_i = AX_i + C + \epsilon_i = f(X_i) + \epsilon_i \quad (4.21)$$

Apart from irreducible error a regression model suffers from a residual error e_i which can be reduced and managed to make it as small as zero or close to zero as indicated in equation (4.22) below. And the irreducible error ϵ_i is part of the residual error e_i in equation (4.21) above.

$$e_i = Y_i - \hat{Y}_i \quad (4.22)$$

Now mean value of all the individual errors ϵ can be made minimum by optimizing the selection algorithm of the slope and the intercept parameters. Therefore, we need to calculate the mean of these errors as shown in equation (4.23) below and then compute its derivative to find out the minima equating it with zero. At the minima or maxima point of any function the slope becomes zero because the straight line passing through these points is horizontal making no angle with x-axis.

$$\text{mean}(e_i) = \frac{1}{N} \sum_{i=1}^N e_i \quad (4.23)$$

But only this cannot guarantee a good model as positive and negative error values could cancel each other and wrongly provide minimum error value. Therefore, it is advisable to focus on either on absolute error or sum of residual error squared value (RSS) as shown in equation (4.24).

$$RSS = \sum_{i=1}^N (e_i)^2 \quad (4.24)$$

Hence, the values of the slope parameter A and the intercept parameter C which offer the minimum RSS value can be obtained using equation (4.25).

$$\arg \min_{A,C} \sum_{i=1}^N (Y_i - AX_i - C)^2 \quad (4.25)$$

To compute the minimum of this RSS value first we take the partial derivative of the RSS function with respect to the parameter C and equate it with zero as shown in equation (4.26).

$$\frac{d(RSS)}{dC} = \frac{d(\sum_{i=1}^N (Y_i - AX_i - C)^2)}{dC} = 0 \quad (4.26)$$

This can be explicitly expressed as shown in equation (4.27).

$$-2 \sum_{i=1}^N (Y_i - AX_i - C) = 0 \quad \gggg \quad \sum_{i=1}^N (Y_i - AX_i) = \sum_{i=1}^N C \quad (4.27)$$

And finally, we compute the optimal intercept parameter as shown in equation (4.28).

$$C = \text{mean}(Y) - A \text{mean}(X) = \bar{Y} - A\bar{X} \quad (4.28)$$

Hence the y-intercept is computed by subtracting a product of the slope A and the average of the input features \bar{X} from the mean of the outcomes \bar{Y} .

The next step is to determine the slope A of the straight line equation. For this purpose we need to take the partial derivative of the RSS equation with respect to (w.r.t) the slope parameter A as in equation (4.29).

$$\frac{d(RSS)}{dA} = \frac{d(\sum_{i=1}^N (Y_i - AX_i - C)^2)}{dA} = 0 \quad (4.29)$$

Similarly this can be expanded as in equation (4.30).

$$-2 \sum_{i=1}^N (Y_i - AX_i - C) X_i = 0 \quad (4.30)$$

Finally, it gives the mathematical relation shown in equation (4.31).

$$\sum_{i=1}^N (Y_i)X_i - A \sum_{i=1}^N X_i X_i - C \sum_{i=1}^N X_i = 0 \quad (4.31)$$

Substituting the expression for C in equation (4.28) above we can compute the formula to determine the slope parameter A following the mathematical relations described in equations.

$$\sum_{i=1}^N (Y_i)X_i - A \sum_{i=1}^N X_i X_i - (\bar{Y} - A\bar{X}) \sum_{i=1}^N X_i = 0 \quad (4.32)$$

$$\sum_{i=1}^N (Y_i)X_i + A(\bar{X} \sum_{i=1}^N X_i - \sum_{i=1}^N X_i X_i) - \bar{Y} \sum_{i=1}^N X_i = 0 \quad (4.33)$$

$$A(\bar{X} \sum_{i=1}^N X_i - \sum_{i=1}^N X_i X_i) = \bar{Y} \sum_{i=1}^N X_i - \sum_{i=1}^N (Y_i)X_i \quad (4.34)$$

$$A = \frac{\bar{Y} \sum_{i=1}^N X_i - \sum_{i=1}^N (Y_i)X_i}{\bar{X} \sum_{i=1}^N X_i - \sum_{i=1}^N X_i X_i} \quad (4.35)$$

$$A = \frac{\frac{1}{N} \sum_{i=1}^N Y_i \sum_{i=1}^N X_i - \sum_{i=1}^N Y_i X_i}{\frac{1}{N} \sum_{i=1}^N X_i \sum_{i=1}^N X_i - \sum_{i=1}^N X_i X_i} = \frac{N\bar{X}\bar{Y} - \sum_{i=1}^N Y_i X_i}{N(\bar{X})^2 - \sum_{i=1}^N (X_i)^2}$$

And finally we can substitute the formula of the slope parameter A obtained in equation (4.35) in to equation (4.28) to compute for the optimal vertical intercept parameter as shown in equation (4.36).

$$C = \bar{Y} - \frac{N\bar{X}\bar{Y} - \sum_{i=1}^N Y_i X_i}{N(\bar{X})^2 - \sum_{i=1}^N (X_i)^2} \quad (4.36)$$

Another fascinating derivation of these regression parameters A and C shown in equations (4.35) and (4.36) respectively above can be done using Bayesian rule by assuming the straight line passes through the origin. Therefore a straight line equation passing through the origin doesn't have a constant term or its intercept is said to be zero. Hence the line equation is expressed as $y_i = wx_i + noise_i$. Here the noise signals are independent, and normal distribution with zero mean and unknown variance of σ^2 . The probability distribution $p(y_i \setminus w, x)$ has a normal distribution with mean wx and variance σ^2 . Having the data points (x_1, y_1) , (x_2, y_2) , (x_3, y_3) , ..., (x_i, y_i) , ..., (x_N, y_N) as evidence we can find out the parameter w using Bayes posterior rules. The posterior distribution is given by: $p(w \setminus x_1, x_2, x_3, \dots, x_N, y_1, y_2, y_3, \dots, y_N)$. This is commonly known as Bayes linear regression [103]. The next step is to work for the maximum likelihood estimation. This includes answering the following questions:

1. For what value of w
 $p(y_1, y_2, y_3, \dots, y_N \setminus x_1, x_2, x_3, \dots, x_N, w)$ is maximized?
2. For what value of w
 $\prod_{i=1}^N p(y_i \setminus x_i, w)$ is maximized?
3. For what value of w
 $\prod_{i=1}^N e^{-\frac{1}{2}(\frac{y_i - wx_i}{\sigma})^2}$ is maximized?
4. For what value of w
 $\sum_{i=1}^N -\frac{1}{2}(\frac{y_i - wx_i}{\sigma})^2$ is maximized ?
5. For what value of w
 $\sum_{i=1}^N (y_i - wx_i)^2$ is minimized ?
6. For what value of w
 $\sum_{i=1}^N (y_i)^2 - 2wx_i y_i + (wx_i)^2$ is minimized?
 $\sum_{i=1}^N (y_i)^2 - 2w \sum_{i=1}^N x_i y_i + (w)^2 \sum_{i=1}^N (x_i)^2$

And this is the final form and easy to see that it has a quadratic form. Therefore the minimum of this function occurs at the bottom of the parabola where the slope of the line tangent to the graph of the function is zero. Meaning the tangent line is horizontal. Hence taking the partial derivative of this function with respect to w and then equating it with zero helps to determine the expression for w value that minimizes the quadratic equation above.

$$-2 \sum_{i=1}^N x_i y_i + 2w \sum_{i=1}^N (x_i)^2 = 0$$

$$w = \frac{\sum_{i=1}^N x_i y_i}{\sum_{i=1}^N (x_i)^2}$$

The regression model presented so far does not show the reality in nature. Outcomes depend on more than one factor in nature. For instance the price of a house depends on the location of the house, the number of rooms, the distance between the nearest supermarkets, the quality of the road connecting it with public centers, the nature of the neighborhood and so on. Some of these factors can be quantized and numerically expressed and others are subjective which cannot be described mathematically. The next section discusses the regression process consisting of multiple input variables (independent variables) determining the outcome.

Multivariate Linear Regression

This is a regression process incorporating multiple features or covariates to determine the outcome. The choice of this regression technique depends on the relationship between each feature and the outcome variable too. As a particular input sample involves vectors or matrix possibly a multivariate regression is carried out using a matrix computation. The multivariate regression model can be approached in the following ways:

Let us assume the following notations for simplicity of our approach

- ✓ Input vector assumed to be $x \in \mathbb{R}^d$
- ✓ Output value assumed to be $y \in \mathbb{R}$
- ✓ Parameters $\beta = (\beta_0, \beta_1, \beta_2, \beta_3, \dots, \beta_d)^T \in \mathbb{R}^{d+1}$
- ✓ Then we set up the model as in equation (4.37)

$$f(x) = \beta_0 + \sum_{j=1}^d \beta_j x_j = x^t \beta \quad (4.37)$$

Given the training data $D = \{(x_i, y_i)\}_{i=1}^N$ the least square cost or loss $L(\beta)$ is defined as in equation (4.38).

$$L(\beta) = \sum_{i=1}^N (y_i - f_i(x))^2 = \sum_{i=1}^N (y_i - x_i^t \beta)^2 = \|Y - X\beta\|^2 \quad (4.38)$$

Here

$$X = \begin{Bmatrix} x_1^t \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ x_N^t \end{Bmatrix} = \begin{Bmatrix} 1, & x_{1,1}, & x_{1,2}, & \dots, & x_{1,d} \\ 1, & x_{2,1}, & x_{2,2}, & \dots, & x_{2,d} \\ \dots & \dots & \dots & \dots & \dots \\ 1, & x_{j,1}, & x_{j,2}, & \dots, & x_{j,d} \\ \dots & \dots & \dots & \dots & \dots \\ 1, & x_{N,1}, & x_{N,2}, & \dots, & x_{N,d} \end{Bmatrix}, \quad Y = \begin{Bmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \\ y_N \end{Bmatrix}$$

$$x_j^t = \{1, x_{j,1}, x_{j,2}, \dots, x_{j,d}\}$$

In addition N and d are the number of samples in our training set and number of features in each sample respectively. To find a minimum loss an optimization technique is applied and optimal parameters β_j that could lead to a linear model are obtained. For this purpose we need to take the first derivative of the loss function in equation (4.39) with respect to β .

$$0_d^t = \frac{\partial L(\beta)}{\partial \beta} = -2(Y - X\beta)^T X \leftrightarrow 0_d^t = X^T X \beta - X^T Y \quad (4.39)$$

Solving for the parameter β gives the equation shown in (4.40).

$$\beta = (X^T X)^{-1} X^T Y \quad (4.40)$$

4.5.2. Non-linear and least square support vector regression (LSSVR)

When a statistical data does not fit in linear models a non-linear model is defined which replace the independent variables x_j in equation (4.37) above with a non-linear function $\varphi(x_j) \in \mathbb{R}^k$ and these functions are named as non-linear features hereafter. The new estimation function $f(x)$ is expressed similarly in equation (4.41) below. Nonlinear regression provides the most flexible curve-fitting functionality. However it can take considerable effort to choose the nonlinear function that creates the best fit for the particular shape of the curve.

$$f(x) = \sum_{j=1}^k \varphi(x_j) \beta_j = \varphi(x)^T \beta \quad (4.41)$$

The expression for optimal parameter β remains the same as shown in equation (4.40) above except the independent variables x_j^T are replaced by non-linear features $\varphi(x_j^T)$. These features depend on the choice of the model attempt to apply. Selected non-linear models will be briefly discussed in this section.

Therefore

$$X = \left\{ \begin{array}{c} \varphi(x_1)^T \\ \cdot \\ \cdot \\ \varphi(x_i)^T \\ \cdot \\ \cdot \\ \varphi(x_N)^T \end{array} \right\} \text{ for non-linear models or kernels}$$

One of the most widely used non-linear models in acoustic modeling is the radial basis function (RBF) which is briefly discussed below. In addition a list of other kernel functions $\varphi(x)$ for univariate dataset is presented below.

Linear : $\varphi(x, \omega) = x^T \omega$

Polynomial : $\varphi(x, \omega) = (\eta + x^T \omega)^d$

Radial basis function (RBF) : $\varphi(x, \omega) = e^{-\frac{\|x-\omega\|^2}{2\sigma^2}}$

Splines : $f(x) = \sum_{j=1}^{m+k+1} \beta_j g_j(x)$, where k is polynomial order, and m is number of polynomial kernel

function $g_j(x)$. The approximation $f(x)$ is a fitting functions while β_j 's are coefficients.

wavelets:

$$W_{ij} = \varphi_j(x_i) \quad \text{where,} \quad x_i = \frac{i}{n}, \quad i = 1, 2, 3, \dots, n$$

String – kernel :

It measures similarity of pairs of strings [32]. Let str1 and str2 be two strings, the kernel $\varphi(\text{str1}, \text{str2})$ would provide a higher value for higher similarity between str1 and str2.

Radial Basis Function (RBF)

Neural Networks are very powerful models for classification tasks. But we used them for regression in our study to develop the least square support vector regression (LSSVR). We used our training dataset and we projected the training trend into the test set to make predictions. Regression has been discussed earlier at the beginning of this section and has many applications in a wide range of areas including finance, physics, medicine, meteorology, biology and many others. Radial basis function (RBF) is a neural network architecture commonly used in non-linear regression as well as function approximation in addition to their popular application in classification [117]. An RBF network is a 2-layer network apart from the output layer. We have an input that is fully connected to a hidden layer. The output of the hidden layer is taken to perform a weighted sum to get our final output. Hence its architecture is not deep. Unlike the neurons in conventional neural networks and deep neural networks (DNN) the neurons in RBF networks contain Gaussian RBF. And hence the Gaussian RBFs are used as the activation functions.

Figure 4.9 below shows some Gaussian densities with different parameters and their combined effect. These Gaussian densities make up the radial basis function. As it can be clearly observed in the figure, the values of individual densities are bound to [0,1]. The resultant density depends on the means and variances of all the individual densities. The individual densities follow normal distribution whose mathematical expression for univariate and multivariate random variables is given by equation (4.42) and (4.43) respectively.

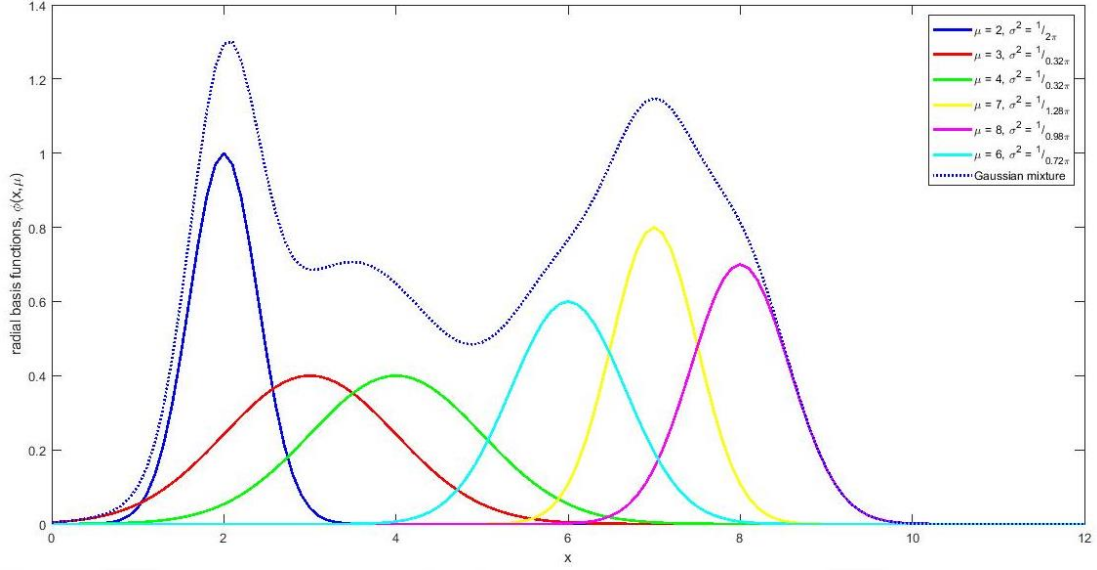


Figure 4.9. Gaussian mixtures and kernel functions

$$N(x|\mu, \sigma) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \quad (4.42)$$

With dimensionality changes on the dataset as well as the parameters modification in equation (4.42) is required in the case of multivariate data. Assuming each sample is d dimensional, equation (4.42) is rewritten as:

$$N(X|\mu, \Sigma) = \frac{e^{-\frac{1}{2}(X-\mu)^T \Sigma^{-1}(X-\mu)}}{\sqrt{(2\pi)^d |\Sigma|}} \quad (4.43)$$

Here $\sqrt{(X-\mu)^T \Sigma^{-1}(X-\mu)}$ is the Mahalanobis distance and $|\Sigma|$ is the determinant of the covariance matrix of the dataset X .

In either univariate or multivariate cases the shape, center and steepness of the bell shaped curve shown in Figure 4.9 above. The mean μ determines the center of the symmetrical graph where half of the whole dataset lays to the left of this vertical line and the other half remains to the right of the symmetrical vertical line representing $(x = \mu)$. In Figure 4.9 above, the Gaussians have different colors and are weighted differently. Taking the sum of all the probability densities gives a continuous function. The parameter which indicates the closeness of individual data sample is the variance σ^2 or in some literatures is the standard deviation σ which is the square root of the variance. Accordingly a large variance shows a wide variation between data samples therefore the resulting bell curve is shorter in height, flat and wide open.

On the other hand a small variance results in a long, steep in shape and indicates very close individual data samples.

Technically, the probability density function (pdf) described in equations (4.34) and (4.35) is used to determine the probability of observing an input x or X in multivariate case given that specific normal distribution. However the bell-curve properties of the Gaussian are more important than the fact that it represents a probability distribution for the application of radial basis function (RBF). It is logical to observe an inverse relation between the maximum of the probability density function which occurs at $(x = \mu)$ and evaluated as $(\sqrt{2\pi\sigma^2})^{-1} = \frac{1}{\sigma\sqrt{2\pi}}$ since the total area covered by the bell curve is supposed to be unity. A linear combination of Gaussian density functions with varied centers and a wide range of variances can be used to approximate any function.

The number of Gaussian density functions needed in function approximation depends on the number of bases or kernels used in our network. The structure of the network is shown in fig. 3 below. K-mean clustering can be used to formulate and place the continuous function created due to the superposition of the individual kernels. The center of each basis function is the means of each respective cluster. The weights $\omega = \{\omega_1, \omega_2, \omega_3, \dots, \omega_K\}$ where K stands for the number of clusters or bases, multiplies the output of the basis functions unlike conventional neural networks in which the weights multiply the input features before computing the activation functions.

The centers c_j for each kernel function $\varphi_j(\cdot)$ of the RBF are determined using k-mean algorithms. The regression process begins with clearly setting the necessary variables and parameters.

The input at the very beginning is a set of features for each sample speaker in our study which is given by $X^t = \{x_1, x_2, x_3, \dots, x_d\}$ where d is the dimension of the input or number of features representing each speaker. The approximation function which produces the estimate age $\hat{Y} = \{\hat{y}_1, \hat{y}_2, \hat{y}_3, \dots, \hat{y}_N\}$ where N represents the number of utterances in the specified dataset, is given by equation (4.44) below:

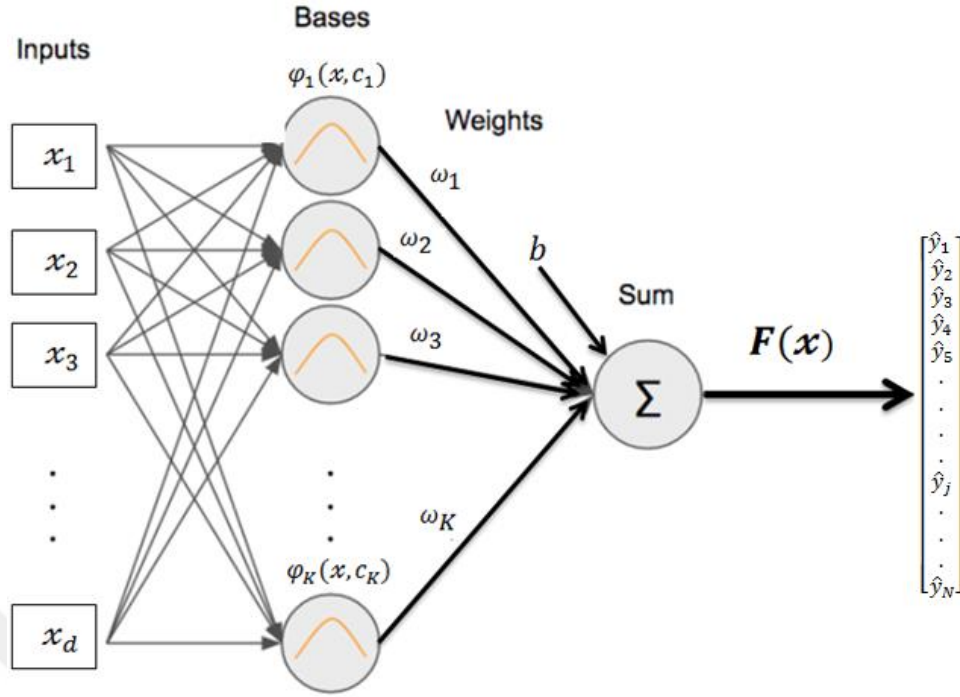


Figure 4.10. Least square support vector regression (LSSVR)

$$F(x) = \sum_{j=1}^K \omega_j \varphi_j(x, c_j) + b \quad (4.44)$$

where ω_j are the weights, b is the bias, K is the number of kernels or clusters or centers of bases in equation (4.44) above. The function approximation is depicted in Figure 4.10 above. The basis functions $\varphi_j(\cdot)$ are the Gaussian RBFs given by equation (4.45).

$$\varphi_j(x, c_j) = e^{-\frac{1}{2\sigma_j^2}(\|x - c_j\|)^2} \quad (4.45)$$

The sum of the squared error between the actual value of individual data points and the values generated by the approximation function is given by the cost formula shown in equation (4.46).

$$error = \sum_{i=1}^N (y^{(i)} - F(x^{(i)}))^2 \quad (4.46)$$

Now we apply optimization algorithms step by step to find optimal weight parameters ω_j and the bias b . For this purpose we take the partial derivative of the error function with respect to ω_j and bias b separately to compute optimal weights and optimal bias in equations (4.47) and (4.48) respectively.

$$\frac{\partial(error)}{\partial \omega_j} = \frac{\partial(error)}{\partial F} \frac{\partial F}{\partial \omega_j} \quad (4.47)$$

$$\frac{\partial(\text{error})}{\partial \omega_j} = \frac{\partial}{\partial F} [\sum_{i=1}^N (y^{(i)} - F(x^{(i)}))^2] \cdot \frac{\partial}{\partial \omega_j} [\sum_{j=1}^K \omega_j \varphi_j(x, c_j) + b] \quad (4.48)$$

The new weights will be updated considering the error they have incurred in the previous iteration using the learning rate η . The result of the partial derivative is given by:

$$\nabla(\text{error}) = -(\sum_{i=1}^N (y^{(i)} - F(x^{(i)}))) \cdot (\sum_{j=1}^K \varphi_j(x, c_j)) \quad \text{Then we deduce the updated weights are } \omega_j \leftarrow \omega_j + \eta(y^{(i)} - F(x^{(i)}))\varphi_j(x, c_j).$$

Similarly for the new bias parameter we take the partial derivative of the error function with respect to b .

$$\begin{aligned} \frac{\partial(\text{error})}{\partial b} &= \frac{\partial(\text{error})}{\partial F} \frac{\partial F}{\partial b} = \frac{\partial}{\partial F} [\sum_{i=1}^N (y^{(i)} - F(x^{(i)}))^2] \cdot \frac{\partial}{\partial b} [\sum_{j=1}^K \omega_j \varphi_j(x, c_j) + b] \\ \frac{\partial(\text{error})}{\partial b} &= (y^{(i)} - F(x^{(i)})) \quad \text{Giving} \quad b \leftarrow b + \eta(y^{(i)} - F(x^{(i)})) \end{aligned}$$

The technique derived to update the weights and the bias parameters is commonly called as the backpropagation in conventional neural networks. This can be converted to pseudo code and eventually to actual code using either python or Matlab. We used Matlab in our experiments [118]. Algorithm 4.1 describes the schematic implementation of LSSVR depicted in Figure 4.10 above.

Algorithm 4.1

- ```

Step .1. Define the radial basis function RBF:
def rbf(x, c, s):
return np.exp(-1 / (2 * s**2) * (x-c)**2)

Step .2. Define the approximation function using superposition of weighted radial basis functions (RBFs)
def predict(self, X):
y_pred = []
for i in range(X.shape[0]):
a = np.array([self.rbf(X[i], c, s) for c, s, in zip(self.centers, self.stds)])
F = a.T.dot(self.w) + self.b
y_pred.append(F)
return np.array(y_pred)

Step .3. Compute the error subtracting values generated by approximation function from actual values

Step .4. Update the weights and bias parameters

Step .5. Continue the process until the error reach a specified level.

```
- 
- 

The perpendicular (the shortest distance between the univariate variable  $x$  and its center  $c_j$  in the exponent of the kernel functions described in equation (4.45) above changes to Mahalanobis distance in the case of multivariate data. Hence the new basis function is given by equation (4.49).

$$\varphi_j(X:\mu) = e^{-\frac{1}{2}(X-\mu_j)^T \Sigma_j^{-1}(X-\mu_j)} \quad (4.49)$$

### 4.5.3. LSTM for speaker age regression

We used LSTM for estimating the age of speakers from MFCC features as well as power spectrum of frames in addition to speaker age classification shown in sub section 5.4.2. As stipulated earlier in this research book at the classifiers list LSTM is suited to classify and predict from patterns based on similarities in a time series data. There exists a cell at the heart of the LSTM network that memorizes previous error values in addition to the three important gates; input, output and forget gates.

Given input features  $x_t$ , the weights  $\{W_f, W_i, W_o \text{ and } W_c\}$  associated with connections from input frames and the weights  $\{V_f, V_i, V_o \text{ and } V_c\}$  associated with connections from the cell at the center to forget, input and output gates as well as the cell itself respectively, the operation of an LSTM network can be described using the following flow diagram and the mathematical expressions are listed below the diagram consecutively. The bias parameters  $b_f, b_i, b_o$  and  $b_c$  in the equations represent the biases directed to forget gate, input gate, output gate and the cell at the center respectively.

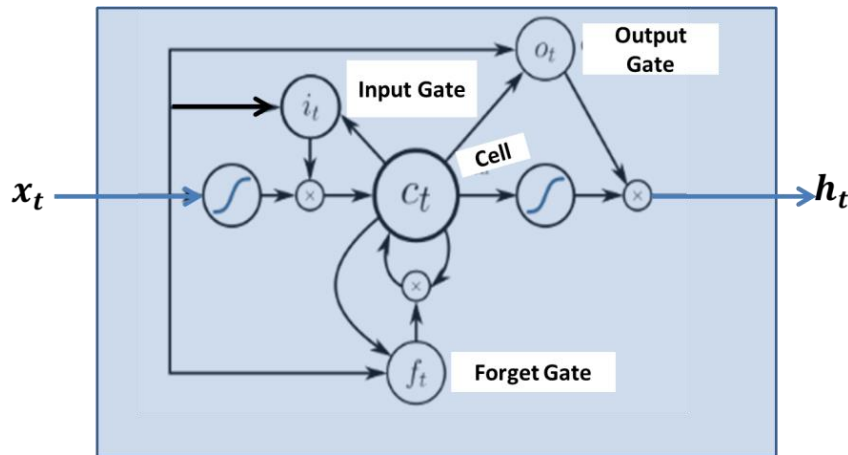


Figure 4.11. Peephole connections in LSTM cells

Figure 4.11 above is commonly referred to as the peephole connection LSTM unit in recurrent neural networks (RNN). Peephole connections stand for the three synapses originating from the cell and terminating at the input, output and forget gates  $i_t, o_t$

and  $f_t$  respectively. The chain of equations (4.50) to (4.55) can be used to determine the outputs of the three gates and the cell shown at the middle of the LSTM.

$$f_t = g_{sigmoid}(W_f x_t + V_f h_{t-1} + b_f) \quad (4.50)$$

$$i_t = g_{sigmoid}(W_i x_t + V_i h_{t-1} + b_i) \quad (4.51)$$

$$o_t = g_{sigmoid}(W_o x_t + V_o h_{t-1} + b_o) \quad (4.52)$$

$$\tilde{c}_t = g_{hyperbolic}(W_c x_t + V_c h_{t-1} + b_c) \quad (4.53)$$

$$c_t = f_t \otimes c_{t-1} + i_t \otimes \tilde{c}_t \quad (4.54)$$

$$h_t = o_t \otimes g_{tanhyperbolic}(c_t) \quad (4.55)$$

The cell contributes the estimated value  $h_{t-1}$  at time step for the current time  $t$  prediction  $h_t$ . The operation  $\otimes$  represents element-wise multiplication between two operands in the above equations. The initial values  $c_0$  and  $h_0$  are assumed to be both zero and the subscripts  $t$  denotes time step.

## 5. EXPERIMENTAL SETUP

### 5.1. Databases

The age and gender annotated telephone speech database (aGender) is exhaustively used to train and test performance of several feature-classifier pairs [14-15, 17, 20, 119]. The database consists of 47 hours of prompted and free text [20]. It includes seven categories: Children (7-14 years old) for both genders, young female (YF, 15-24 years old), young-male (YM, 15-24 years old), adult-female (AF, 25-54 years old), adult-male (AM, 25-54 years old), senior-female (SF, 55-80 years old), and senior-male (SM, 55-80 years old). Due to commercial concerns and considering that it is easier to classify compared to the other classes, most studies avoid the children class in their researches [17, 60, 67, 119]. Children's speech both in male and female contains relatively higher fundamental frequency  $F_0$  which makes it easily separable and more classifiable compared to young, adult and senior utterances [28]. Therefore, we limited our focus on speakers older than 15 years in order to compare our approaches with past and ongoing studies. Young, adult and senior classes are treated separately in their respective genders. This database is prepared to help in overcoming the low compatibility of results, by addressing three selected sub-challenges namely; age, gender and affect sub-challenges [14]. A total of 184 male and 190 female speakers are used in the training set. The development set consists of 130 male and 131 female speakers. 15 male and 14 female speakers are selected for testing performance as summarized in table 5.1 below.

Table 5.1. Distribution of speakers along development, training and test sets in each class of the aGender database.

|        | Development set | Training set | Test set | Total |
|--------|-----------------|--------------|----------|-------|
| Female | 131             | 190          | 14       | 335   |
| Male   | 130             | 184          | 15       | 329   |

The audios in the aGender database were recorded over cell phones and landline connections in 8000 Hz, 8 bit alaw format. The male and female datasets are further classified in to three categories as young (ages: 15-24), adult (ages: 25-54) and old or

senior (ages: 55-80). A total of 852 German speakers (at least 100 speakers in each class) have participated in the audio recording which accounts for 47 hours of speech [120]. All the seven classes including children class ranging from age 7 to 14 are considered to evaluate the overall performance of fusion of features in one scenario [76]. The distribution of utterances in each class for development, training and test sets is presented in table 5.2 below. Due to the lack of labeling on the test set of the original dataset we received the training dataset is split in to training and test sets. Once the path to each speech utterance in the three datasets is established Matlab commands are exploited to trace and pick for processing. A total of 9644, 12985 and 1150 audio samples are used as development, training and test sets in the female speaker age classification experiments respectively. Similarly 8508, 12906 and 1079 utterances are used in the male gender respectively.

Table 5.2. Distribution of utterances along development, training and test sets in each class of the aGender database.

| Age Classes  | Development set | Training set | Test set | Total |
|--------------|-----------------|--------------|----------|-------|
| Child 7-14   | 2397            | 4000         | 407      | 6804  |
| Female 15-24 | 2722            | 4254         | 384      | 7360  |
| Female 25-54 | 3361            | 4187         | 386      | 7934  |
| Female 55-80 | 3561            | 4544         | 380      | 8485  |
| Male 15-24   | 2170            | 3631         | 388      | 6577  |
| Male 25-54   | 2512            | 4051         | 366      | 7295  |
| Male 55-80   | 3826            | 5224         | 325      | 9700  |
| Female Total | 9644            | 12985        | 1150     | 23779 |
| Male Total   | 8508            | 12906        | 1079     | 22493 |
| Grand Total  | 20549           | 29891        | 2636     | 53076 |

Age-Vox-Celeb database consists of YouTube recordings of celebrities. For this reason, it contains a large number of utterances in the adult class for both genders. This class is well represented in terms of speaker and utterance diversity. However, young and senior classes lack this luxury [21, 70-71]. Table 5.3 below shows the distribution of utterances in this database for the three speaker age classes.

Table 5.3. Distribution of utterances along development, training and test sets in each class of the Age-Vox-Celeb database.

| Age Classes  | Development set | Training set | Test set | Total |
|--------------|-----------------|--------------|----------|-------|
| Female 15-24 | 1820            | 3710         | 699      | 6229  |
| Female 25-54 | 2594            | 6738         | 2000     | 11332 |
| Female 55-80 | 2810            | 3770         | 999      | 7579  |
| Male 15-24   | 2549            | 4013         | 900      | 7462  |
| Male 25-54   | 3526            | 7497         | 1209     | 12232 |
| Male 55-80   | 3857            | 7022         | 1200     | 12079 |
| Female Total | 7224            | 14218        | 3698     | 25140 |
| Male Total   | 9932            | 18532        | 3309     | 31773 |
| Grand Total  | 17156           | 32750        | 7007     | 56913 |

We had fewer speakers and more number of utterances in the Turkish database compared to aGender and Age-Vox-Celeb. The aGender is the most balanced among the three databases. Each speaker contributed several utterances in the Turkish database although their number was quite a few. The sampling rate used in the Turkish and English databases is 16 kHz. The details of the Turkish database are provided in table 5.3.

Table 5.4. Distribution of speakers along development, training and test sets in each class of the Turkish database.

| Age Classes   | Development set | Training set | Test set | Total |
|---------------|-----------------|--------------|----------|-------|
| Female 15-25  | 6               | 10           | 41       | 57    |
| Female 26-40  | 6               | 10           | 107      | 113   |
| Female 41-100 | 6               | 10           | 33       | 49    |
| Male 15-25    | 6               | 10           | 31       | 47    |
| Male 25-40    | 6               | 10           | 77       | 93    |
| Male 41-100   | 6               | 10           | 0        | 16    |
| Female Total  | 18              | 30           | 181      | 229   |
| Male Total    | 18              | 30           | 108      | 156   |
| Grand Total   | 36              | 60           | 288      | 384   |

## 5.2. Classification and Regression Experimental Setups

All our experiments begin with establishing well organized databases to each audio sample. Then our subroutines pick the sample utterances from this path before it commences to other operations as shown in Figure 5.1 below. Details are presented in the experimental setup section mean while our databases are organized in gender, age classes, training, test and development (UBM) sets.

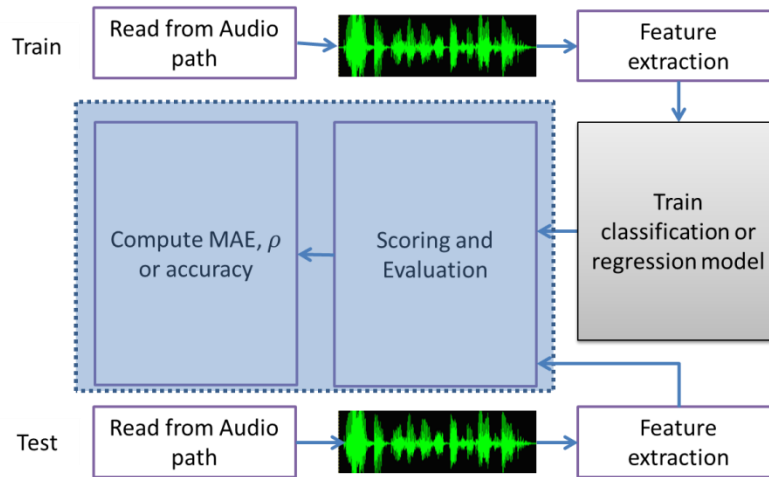


Figure 5.1. General block diagram that shows the overview of experiments in this study

A hamming window of length 20 ms with 10 ms overlap is used for framing utterances [121]. 512 DFT point and a total variability dimension of 200 are applied. 13 static, 13 dynamic and 13 acceleration features are extracted from each frame. This makes up a total of 42 features including an energy component for each of the three feature sets.

For speech duration analysis we used the setup displayed below.

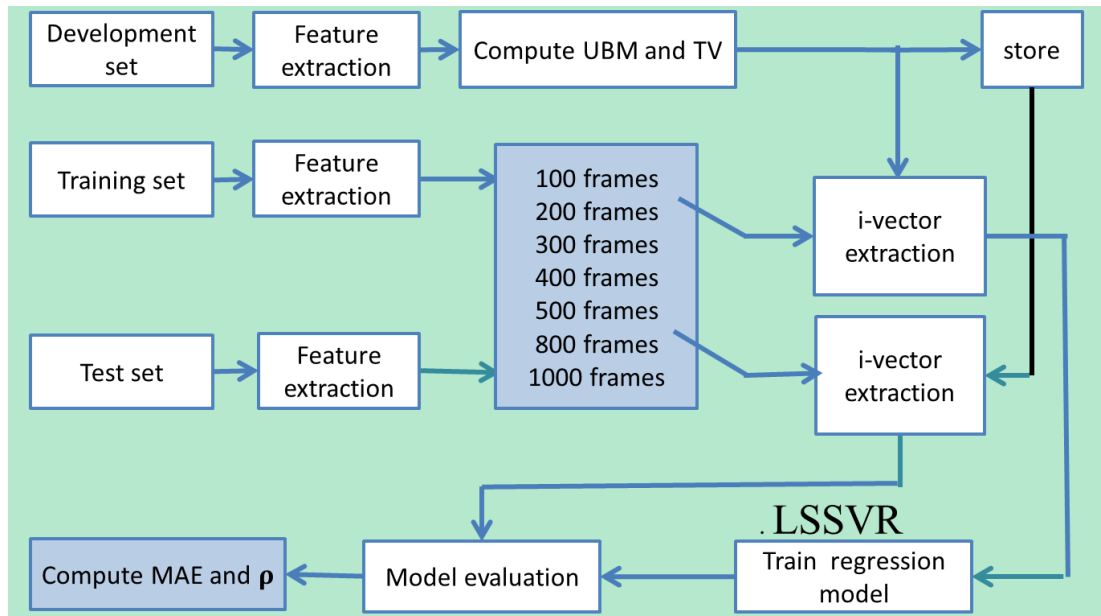


Figure 5.2. Speech length in terms of number of frames for age estimation



After extracting the acoustic features we stored these values in a matrix variable and adjust the duration according to selected sizes from the shortest being 0.5 second (50 frames) long and the longest as 10 seconds (1000 frames) long.

Several network architectures of DNN and LSTM have been attempted in order to raise the accuracy of prediction and reduce the mean absolute error. However, DNN failed to deliver up to our expectation. On top of that it takes longer to complete even 20 epochs. LSTM on the other hand, has a comparable performance in the female aGender dataset with MFCC feature and the best end-to-end prediction accuracy compared to CDS, GMM and PLDA with this dataset. The network setup includes 7 layers; an input layer with input dimension 42, three bidirectional LSTM layers with 128, 32 and 32 hidden units (neurons) respectively, a fully connected layer with 3 neurons, a softmax layer and an output layer with cross-entropy. Sigmoid activation function is applied in most of the neurons. For end-to-end setting, the same number of layers is used with the same hidden layers except for the input dimension which is increased to 257. The input dimension is determined from the DFT point  $N$  we applied in the FFT algorithm. Hence, the input dimension is half of the DFT point  $N$  plus 1. Stochastic gradient descent (SGD), a maximum epoch of 20 and minimum batch size of 27 are implemented.

Two metrics are used to evaluate the performance of the LSSVR and LSTM regression algorithms. The majority of our experiments have dealt with speaker age classification rather than regression especially at the beginning of our research. But later regression approaches are included for speaker age estimation. The famous mean absolute error (MAE) and Pearson correlation coefficient ( $\rho$ ) are used to evaluate the performance of linear and non-linear regression approaches. These two parameters are defined using actual and estimated speaker age data as shown in equation (5.1) and (5.2) below.

$$MAE = \frac{1}{N} \sum_{i=1}^N |y_i - \tilde{y}_i| \quad (5.1)$$

$$\rho = \frac{1}{N-1} \sum_{i=1}^N \left( \frac{y_i - \mu_y}{\sigma_y} \right) \left( \frac{\tilde{y}_i - \mu_{\tilde{y}}}{\sigma_{\tilde{y}}} \right) \quad (5.2)$$

Where  $y_i$  and  $\tilde{y}_i$  are the actual and estimated age of the  $i^{th}$  utterance respectively. In addition  $\mu_y$  and  $\sigma_y$  are the mean and standard deviation for the predicted ages of

the test set; and  $\mu_y$  and  $\sigma_y$  for the true ages. Higher correlation coefficients and lower mean absolute error values are better. The performance evaluation of our approaches is presented in section 5 below.

Similar approaches are followed in pre-processing of the audio signal during feature extraction. However, instead of writing the features on to a text file we created a matrix that accumulates all the features in each frame immediately after extraction is completed. The usual hamming window of size 20 milliseconds and 10 milliseconds overlap is applied for framing each audio file [17]. Most of the features we used and designed are inspired by an article that investigated several features (magnitude as well as phase-based) for replay spoofing attack detection [18]. We have also contributed a new feature set called parabolic filter mel-frequency cepstral coefficient (PFMFCC). Our experimental result for the new feature is published in [76].

For speaker age classification schemes, accuracy is used as a measure of performance metrics in our experimental results. The usual way to do this is to generate the confusion matrix. This matrix is a square matrix, whose order is equivalent to the number of classes in a classification problem, consisting of correct and misclassified number of utterances. The rows in a confusion matrix stand for actual classes whereas the columns represent predicted classes. The elements placed along the diagonals convey the number of correctly classified utterances for each class as it goes down the diagonal. On the other hand, those elements found off the diagonal represent the number of utterances wrongly classified. The misclassification mostly occurs in the adult speakers in large numbers; misclassified as either young or senior (old) speakers as it shares boundary with both classes. Hence the accuracy can be computed as the average of the elements along the diagonal if the classes are evenly distributed. If there are irregularities in the number of utterances in each class, then the accuracy is computed as the sum of correctly classified utterances divided by the total number of test set utterances given by the equation (5.3) below:

$$Accuracy = \frac{\#correctly\ classified\ utterances}{\#test\ set\ utterances} \quad (5.3)$$

where the hashtag # represents number of something.

## **6. RESULTS AND DISCUSSION**

This chapter is presented in 2 subsections. The first one presents experimental results with using tables and figures. Short discussions are provided right after each table and figure. The second subsection gives in-depth analysis and discussion on selected results and unexpected outcomes.

### **6.1. Results**

Experimental results of the research study are presented graphically as well as using tables and a short discussion follows in this subsection. However, in-depth analysis and discussion is provided in section 6.2. The results are presented in 4 categories; the first category is matched-language scenario carried out on German and Turkish databases, secondly performance evaluation of bilingual, multilingual and cross-language evaluation results which consists of utterances from English, German and Turkish languages, thirdly regression results for the least square support vector regression (LSSVR) model and speech length (duration) analysis for selected feature sets and finally performance evaluation of deep learning based classification models.

#### **6.1.1. Performance evaluation of CDS, GMM and PLDA classifiers on matched-language baseline scenarios**

This subsection presents the performance evaluation of three classifiers and ten feature sets for the German and Turkish databases. Table 6.1 presents speaker age classification performance in terms of accuracy for; CDS, GMM and PLDA classifiers over the female dataset of aGender database by applying VAD with an energy threshold of -55dB to remove non-speech and silent frames. If the maximum energy among all frames is greater than -25dB, the threshold is raised to 30dB below the maximum energy. The former follows absolute criteria whereas the latter uses a relative approach to remove non-speech frames. The table also presents evaluation results for non-VAD scenarios in which silence and noise frames are kept.

Table 6.1. Comparing the proposed PFMFCC in female datasets of the aGender database with and without VAD

| a) Female without VAD |           | Accuracies in % |               |               |
|-----------------------|-----------|-----------------|---------------|---------------|
|                       |           | CDS             | GMM           | PLDA          |
| Feature sets          | MFCC      | 56.436          | 57.033        | 57.033        |
|                       | IMFCC     | 46.632          | 43.904        | 50.127        |
|                       | LFCC      | 52.783          | 57.739        | 52.956        |
|                       | RFCC      | 53.565          | <b>58.522</b> | 53.367        |
|                       | PFMFCC    | 50.440          | 52.740        | <b>58.140</b> |
|                       | SCMF      | 51.321          | 52.173        | 51.321        |
|                       | Cosine_ph | 53.623          | 49.616        | 44.245        |
|                       | SSFC      | 52.429          | 50.639        | 51.15         |
|                       | MODGD     | <b>56.947</b>   | <b>57.971</b> | 50.724        |
|                       | RASTA-PLP | 46.717          | 53.367        | 55.413        |
| b) Female with VAD    |           | Accuracies in % |               |               |
|                       |           | CDS             | GMM           | PLDA          |
| Feature sets          | MFCC      | 56.265          | <b>57.033</b> | 46.973        |
|                       | IMFCC     | 53.793          | 51.577        | 29.497        |
|                       | LFCC      | 52.941          | 52.429        | 42.966        |
|                       | RFCC      | 54.305          | 51.832        | 44.330        |
|                       | PFMFCC    | 50.350          | 55.390        | <b>52.170</b> |
|                       | SCMF      | <b>57.033</b>   | 53.623        | 44.586        |
|                       | Cosine_ph | 47.058          | 44.842        | 39.471        |
|                       | SSFC      | 53.878          | 51.065        | 38.704        |
|                       | MODGD     | 50.895          | 50.127        | 35.720        |
|                       | RASTA-PLP | 47.826          | 50.639        | 47.996        |

A maximum of 57.03% accuracy using SCMF on CDS and MFCC on GMM classifier is achieved applying voice activity detection (VAD)[122] for all feature extractions over the female dataset of the aGender database. However, PLDA generally delivered poor performances in this regard. It offered accuracies below 50% with all feature sets except PFMFCC which showed 52.17% and 51.3% accuracies in correctly predicting the age classes for female and male test samples respectively. Similarly a maximum of 47.729% accuracy using RFCC on cosine scoring, 47.358% using RASTA-PLP on GMM classifier and 46.987% accuracy using IMFCC on PLDA classifier is achieved for male dataset. Table 6.2 below presents speaker age performances of the three classifiers; CDS, GMM and PLDA over male dataset of the aGender database using the 10 feature sets discussed in chapter 2. The table presents for both with VAD and without VAD scenarios.

Table 6.2. Comparing the proposed PFMFCC in male datasets of the aGender database with and without VAD

| a) Male without VAD |               | Accuracies in % |               |               |
|---------------------|---------------|-----------------|---------------|---------------|
|                     |               | CDS             | GMM           | PLDA          |
| Feature sets        | MFCC          | 41.797          | 42.632        | 55.144        |
|                     | IMFCC         | 41.149          | 40.963        | 54.587        |
|                     | LFCC          | 47.544          | 46.339        | 52.641        |
|                     | RFCC          | 43.837          | 40.963        | 56.348        |
|                     | <b>PFMFCC</b> | <b>51.060</b>   | <b>56.010</b> | <b>57.230</b> |
|                     | SCMF          | 43.466          | 41.334        | 52.641        |
|                     | Cosine_ph     | 39.295          | 33.271        | 31.417        |
|                     | SSFC          | 37.256          | 37.905        | 48.656        |
|                     | MODGD         | 46.153          | 51.159        | 45.319        |
|                     | RASTA-PLP     | 43.188          | 40.685        | 48.285        |
| b) Male with VAD    |               | Accuracies in % |               |               |
|                     |               | CDS             | GMM           | PLDA          |
| Feature sets        | MFCC          | 43.929          | 42.354        | 43.651        |
|                     | IMFCC         | 40.871          | 41.797        | 46.987        |
|                     | LFCC          | 42.354          | 42.910        | 39.851        |
|                     | RFCC          | <b>47.729</b>   | 41.705        | 38.832        |
|                     | <b>PFMFCC</b> | 45.320          | 42.170        | <b>51.300</b> |
|                     | SCMF          | 45.968          | 44.763        | 45.783        |
|                     | Cosine_ph     | 35.866          | 33.827        | 39.202        |
|                     | SSFC          | 37.998          | 39.573        | 33.456        |
|                     | MODGD         | 36.793          | 35.495        | 40.500        |
|                     | RASTA-PLP     | 42.724          | <b>47.358</b> | 42.910        |

Without VAD, CDS classifier offered 56.95% with MODGD feature and 51.06% with the PFMFCC feature for female and male test sets respectively. Similarly the GMM classifier achieved a maximum of 58.52% and 56.01% with RFCC and PFMFCC features for female and male test sets respectively. The phase-based feature MODGD also impressed with accuracies of 57.97% and 51.16% using this classifier over female and male datasets respectively. These results are shown in Tables 6.1 and 6.2 as well as in Figures 6.1 and 6.2 for female and male test sets with and without VAD respectively. PLDA performed better without VAD in all the features except the cosine phase feature for male test set. The best performances without VAD for this classifier are 58.14% and 57.23% using PFMFCC for female and male test sets respectively. Figures 6.1 and 6.2 below present the results graphically for female and male test sets respectively in order to make the comparisons visually clear.

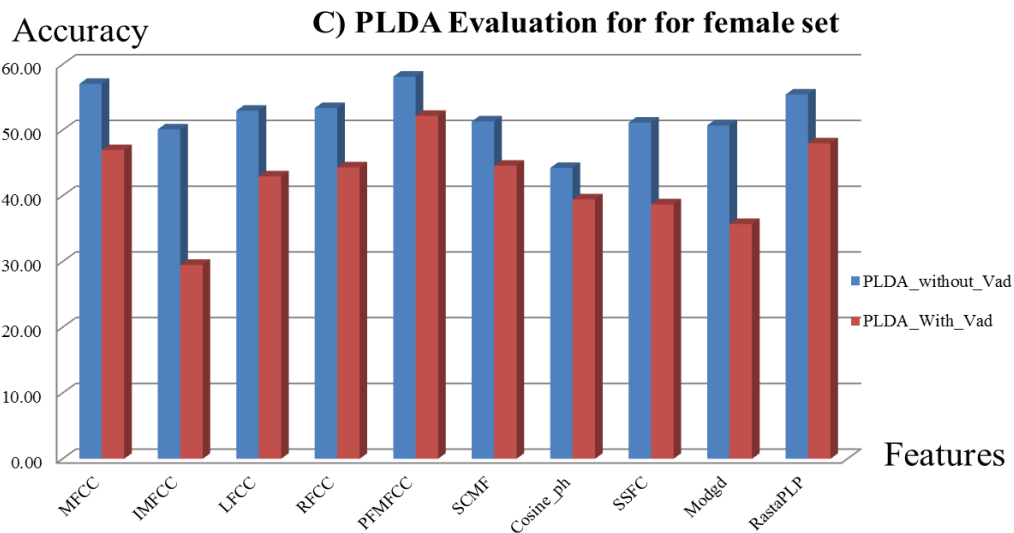
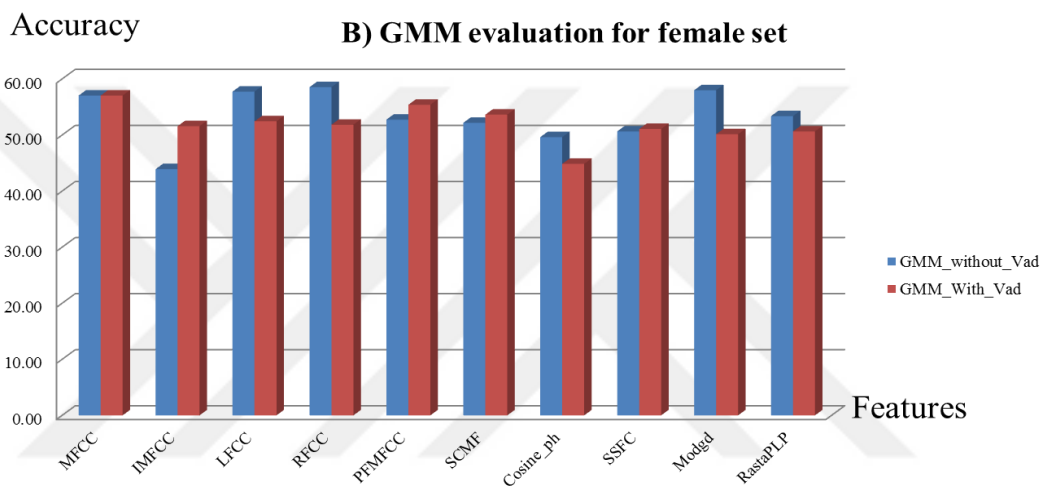
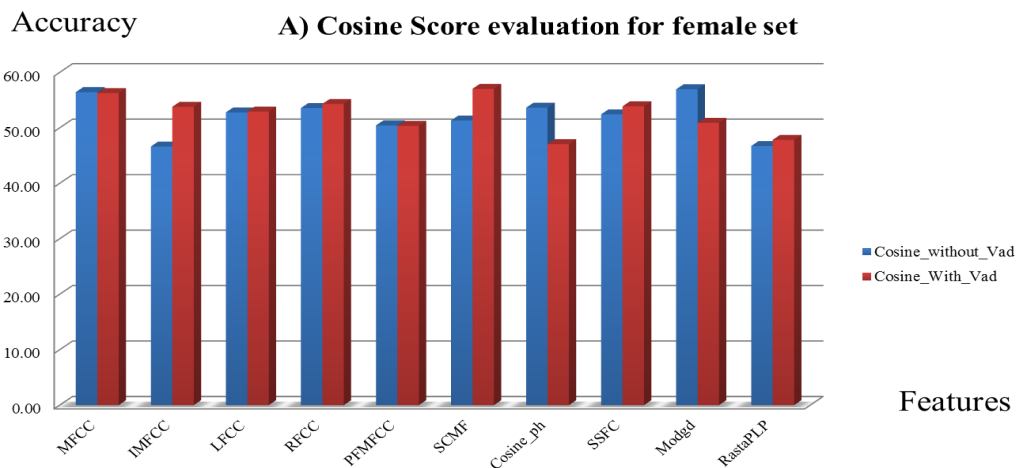


Figure 6.1. Graphic representation of evaluation results for female test set in a) cosine score b) GMM and c) PLDA classifiers with aGender database

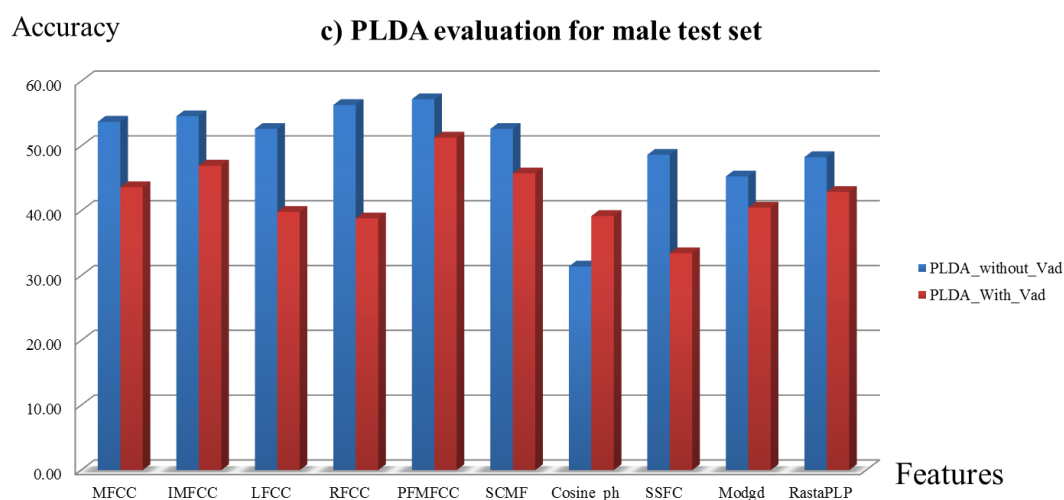
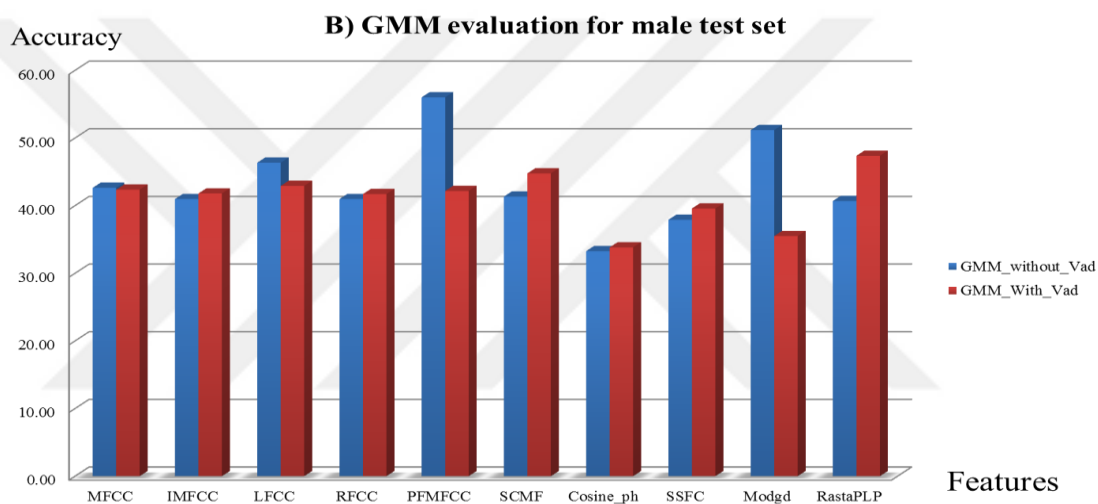
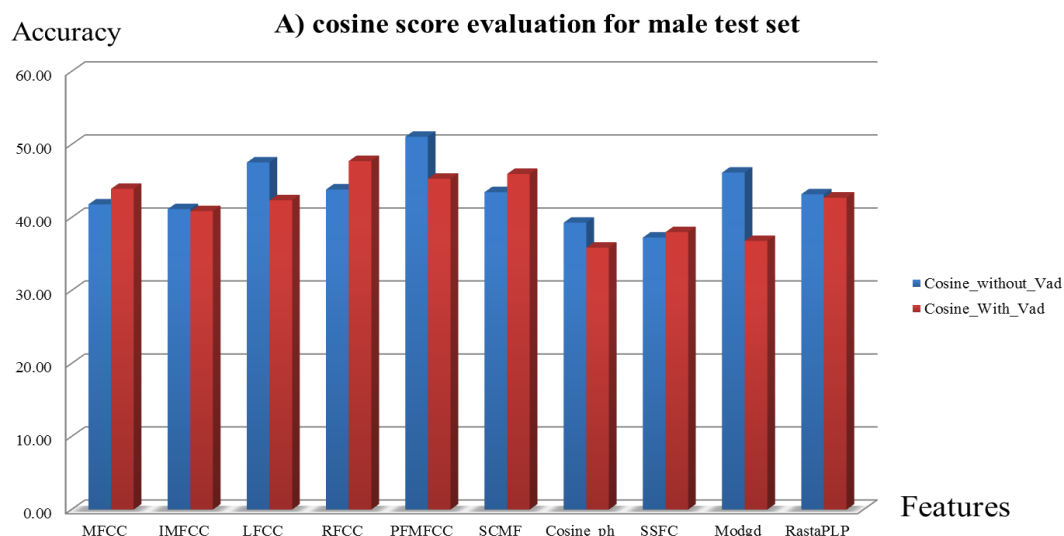


Figure 6.2. Graphic representation of evaluation results for male test set in a) cosine score b) GMM and c) PLDA classifiers with aGender database

In summary, the proposed PFMFCC and the adopted RFCC feature sets offered the best performances with PLDA and GMM classifiers for male and female datasets

respectively. They offered 57.23% and 58.52% correct predictions respectively. PFMFCC also offered 58.14% accuracy with PLDA classifier for the female dataset. A brief comparison of this feature set with MFCC is presented in section 6.2.

The male dataset of the Turkish database generally showed poor performance in all setups comparatively. This is obviously because of imbalance in the database where it presents a large number of adults but, relatively very few senior male speakers. This has also been observed in a previous study with the same database where the confusion matrix shows 0 correct classifications for senior male speakers [53]. The female dataset on the other hand, performed well as it constitutes diversity not only in utterances and speakers but also in age classes comparatively in a balanced manner. Selected experimental results are depicted in Table 6.3 for both genders of the Turkish database.

Table 6.3. Comparing the proposed PFMFCC for female and male datasets in the Turkish database

|              |        | Female accuracies in % |              |              | Male accuracies in % |              |              |
|--------------|--------|------------------------|--------------|--------------|----------------------|--------------|--------------|
|              |        | CDS                    | GMM          | PLDA         | CDS                  | GMM          | PLDA         |
| Feature sets | MFCC   | 62.24                  | 70.45        | <b>51.98</b> | <b>47.44</b>         | 34.16        | 47.56        |
|              | LFCC   | <b>64.70</b>           | 70.42        | 29.29        | 42.31                | 37.46        | 28.76        |
|              | RFCC   | 62.98                  | <b>70.60</b> | 30.83        | 47.19                | <b>41.65</b> | 49.42        |
|              | PFMFCC | 57.18                  | 69.50        | 49.30        | 44.12                | 38.74        | 49.85        |
|              | MODGD  | 36.14                  | 36.47        | 34.75        | 37.04                | 37.68        | <b>50.25</b> |

### 6.1.2. Performance evaluation of CDS, GMM and PLDA classifiers on bilingual, multilingual and cross-language scenarios

The essence of this study is to find out the effect of language in speaker age estimation along with other factors such as number of speech frames involved in training and test sets. Table 6.4 presents experimental results of multilingual (bilingual) training setup for speaker age classification tested with female and male datasets of the German (aGender) and Turkish databases. This table presents performance evaluation of the three models; CDS, GMM and PLDA trained with data composed of audio data in German and Turkish languages.



Table 6.4. Bilingual training tested with German and Turkish female and male datasets for speaker age classification

|                    |        | Test sets (accuracies in %) |              |              |         |              |              |
|--------------------|--------|-----------------------------|--------------|--------------|---------|--------------|--------------|
|                    |        | German                      |              |              | Turkish |              |              |
|                    |        | CDS                         | GMM          | PLDA         | CDS     | GMM          | PLDA         |
| <b>a) Female</b>   |        |                             |              |              |         |              |              |
| Bilingual Training | MFCC   | 37.22                       | 55.22        | 47.65        | 62.24   | 70.45        | 48.94        |
|                    | LFCC   | 54.26                       | 52.17        | 37.3         | 65.03   | 70.16        | 48.68        |
|                    | RFCC   | 36.00                       | <b>57.30</b> | 35.48        | 62.68   | <b>70.93</b> | 44.87        |
|                    | PFMFCC | 44.00                       | 54.00        | 45.65        | 54.84   | 68.99        | 49.23        |
|                    | MODGD  | 35.65                       | 36.09        | 32.35        | 36.03   | 34.53        | 21.59        |
| <b>b) Male</b>     |        |                             |              |              |         |              |              |
| Bilingual Training | MFCC   | 35.22                       | 30.31        | <b>40.41</b> | 48.52   | 32.37        | <b>56.43</b> |
|                    | LFCC   | 33.64                       | 33.18        | 34.66        | 45.69   | 37.46        | 30.02        |
|                    | RFCC   | 34.57                       | 32.44        | 30.95        | 47.57   | 41.65        | 49.42        |
|                    | PFMFCC | 34.29                       | 36.70        | 37.81        | 44.12   | 38.74        | 51.13        |
|                    | MODGD  | 35.87                       | 30.03        | 35.77        | 37.04   | 37.68        | 51.48        |

The multi-language training with only German and Turkish utterances a.k.a. bilingual scenario degraded the matched performance significantly for the aGender database with few exceptions in the GMM classifier over the female dataset. Notable deficits in accuracy include 28.57% and 25.4% decline with PLDA classifier applied on PFMFCC and RFCC features for female and male datasets respectively. This could partly be due to the differences in an audio recording devices and sampling rate. The German audios are recorded from telephone conversations with a sampling rate of 8 kHz whereas the Turkish utterances are recorded with a computer at a sampling rate of 16 kHz. More discussion on what caused these degradations is provided in subsection 6.2 and compares it with multilingual for three language and cross-language scenarios. On the other hand, the performance remained not much affected on the Turkish database compared to matched-language scenarios. The PLDA classifier performed even better with MFCC, PFMFCC and MODGD feature sets especially for the male gender. This could possibly be due to the nature of phoneme sequences in the training utterances that made these features more classifiable than others with the PLDA classifier. It indicates addition of German utterances to the training has contributed for the performance improvement for the PLDA classifier with these features.

Effect of adding a third language to the multi-language (multilingual) training setup is investigated and the results are presented in Table 6.5 below. In this table the

columns represent the language of test sets used for performance evaluation whereas, the training constitutes three languages; German from the aGender database, Turkish and English from Age-Vox-Celeb database.

Table 6.5. Multi-language training performance evaluation for female and male datasets with German (aGender), Turkish and English (Age-Vox-Celeb) databases

| a) Female datasets    |        | Test sets (accuracies in %) |              |       |              |              |       |         |              |       |
|-----------------------|--------|-----------------------------|--------------|-------|--------------|--------------|-------|---------|--------------|-------|
|                       |        | German                      |              |       | Turkish      |              |       | English |              |       |
|                       |        | CDS                         | GMM          | PLDA  | CDS          | GMM          | PLDA  | CDS     | GMM          | PLDA  |
| Multilingual Training | MFCC   | 44.96                       | 46.17        | 35.22 | 65.28        | 65.07        | 47.98 | 36.93   | 42.67        | 36.58 |
|                       | LFCC   | 35.13                       | 48.17        | 31.57 | 63.75        | 70.12        | 30.06 | 29.76   | 42.42        | 31.71 |
|                       | RFCC   | 30.61                       | <b>49.04</b> | 30    | 64.08        | <b>70.13</b> | 27.97 | 30.79   | 42.58        | 31.00 |
|                       | PFMFCC | 39.74                       | 47.22        | 34.52 | 65.10        | 66.31        | 50.26 | 31.27   | <b>45.92</b> | 36.17 |
|                       | MODGD  | 32.17                       | 34.52        | 34.17 | 47.51        | 53.08        | 27.46 | 34.55   | 26.56        | 22.48 |
| b) Male datasets      |        |                             |              |       |              |              |       |         |              |       |
| Multilingual Training | MFCC   | 37.16                       | 35.59        | 32.34 | <b>54.10</b> | 32.47        | 44.88 | 36.50   | 44.85        | 42.13 |
|                       | LFCC   | 33.73                       | <b>39.48</b> | 29.19 | 51.22        | 44.00        | 38.50 | 37.53   | 44.76        | 43.25 |
|                       | RFCC   | 32.07                       | 38.27        | 34.94 | 53.65        | 43.59        | 37.56 | 38.99   | 44.33        | 43.67 |
|                       | PFMFCC | 35.03                       | 36.89        | 29.84 | 39.31        | 42.03        | 35.84 | 36.27   | <b>47.54</b> | 37.29 |
|                       | MODGD  | 31.88                       | 31.05        | 26.14 | 45.75        | 36.41        | 45.07 | 34.36   | 36.66        | 36.81 |

Adding English to the multi-language scenario has improved performance of some feature sets on certain classifiers for some datasets notably; MFCC and RFCC features on CDS classifier for male Turkish and German dataset. In addition, the GMM classifier showed a slight improvement in LFCC, RFCC and PFMFCC feature sets. Significant increase in accuracy is scored for all the three classifiers on MFCC feature over the German female dataset. Likewise, the MODGD feature also offered significant improvement for the female dataset of the Turkish dataset with addition of the Age-Vox-Celeb database to the multilingual setup. Major degradations in performance most likely due to addition of the English dataset occur in the PLDA classifier for the male Turkish dataset with the exception of the LFCC feature which improved the prediction accuracy by 4.87% in this regard.

On the contrary to the multilingual scenarios, language mismatches between training and test sets have been investigated to degrade the performance dramatically and results are presented in Table 6.6 below. In this table the cells located along the diagonal are accuracies of matched-language setups whereas, the cells off-diagonal are language mismatch (cross-language) performance evaluations the rows and columns being training and test sets respectively. The performance has been affected

dramatically for language mismatches in most of the cases compared to matched-language and multilingual scenarios. However, quite few evaluations surprisingly showed better performances than matched-language scenarios.

Table 6.6. Cross-language and matched language performance evaluation for female and male datasets trained with German, Turkish and English databases tested with German, Turkish and English test sets

| a) Female datasets |        | Test sets (accuracies in %) |       |       |         |       |       |         |       |       |
|--------------------|--------|-----------------------------|-------|-------|---------|-------|-------|---------|-------|-------|
|                    |        | German                      |       |       | Turkish |       |       | English |       |       |
|                    |        | CDS                         | GMM   | PLDA  | CDS     | GMM   | PLDA  | CDS     | GMM   | PLDA  |
| German Training    | MFCC   | 56.44                       | 57.03 | 57.03 | 53.41   | 23.64 | 25.55 | 38.18   | 41.34 | 24.13 |
|                    | LFCC   | 52.78                       | 57.74 | 52.96 | 41.13   | 37.72 | 19.90 | 23.30   | 42.02 | 24.32 |
|                    | RFCC   | 53.57                       | 58.52 | 53.37 | 51.98   | 26.43 | 31.67 | 26.79   | 33.01 | 27.62 |
|                    | PFMFCC | 50.44                       | 52.74 | 58.14 | 44.39   | 53.01 | 21.19 | 39.39   | 32.90 | 23.57 |
|                    | MODGD  | 56.95                       | 57.97 | 50.72 | 45.05   | 52.79 | 45.01 | 32.77   | 37.96 | 25.49 |
| Turkish Training   | MFCC   | 34.96                       | 35.65 | 30.17 | 62.24   | 70.45 | 51.98 | 24.68   | 21.65 | 19.48 |
|                    | LFCC   | 31.57                       | 33.57 | 33.48 | 64.70   | 70.42 | 29.29 | 20.73   | 20.08 | 20.27 |
|                    | RFCC   | 30.61                       | 33.57 | 33.48 | 62.98   | 70.60 | 30.83 | 20.02   | 19.89 | 20.13 |
|                    | PFMFCC | 34.35                       | 37.04 | 30.61 | 57.18   | 69.50 | 49.30 | 21.35   | 22.73 | 19.67 |
|                    | MODGD  | 34.61                       | 32.78 | 34.26 | 36.14   | 36.47 | 34.75 | 28.73   | 24.27 | 23.00 |
| English Training   | MFCC   | 33.56                       | 29.22 | 25.91 | 41.83   | 37.94 | 57.63 | 43.26   | 45.81 | 45.86 |
|                    | LFCC   | 32.09                       | 26.00 | 31.57 | 34.71   | 40.83 | 52.02 | 39.23   | 42.91 | 43.70 |
|                    | RFCC   | 30.52                       | 24.00 | 26.43 | 37.87   | 40.84 | 53.56 | 37.58   | 43.37 | 43.59 |
|                    | PFMFCC | 32.69                       | 24.78 | 25.30 | 43.51   | 28.99 | 60.56 | 42.94   | 46.27 | 45.78 |
|                    | MODGD  | 34.17                       | 32.61 | 33.22 | 35.23   | 25.51 | 42.89 | 32.79   | 35.42 | 39.04 |
| b) Male datasets   |        |                             |       |       |         |       |       |         |       |       |
| German Training    | MFCC   | 41.79                       | 42.63 | 55.14 | 25.72   | 30.24 | 29.80 | 40.62   | 40.47 | 35.60 |
|                    | LFCC   | 47.54                       | 46.34 | 52.64 | 22.32   | 13.43 | 34.92 | 34.48   | 38.29 | 36.41 |
|                    | RFCC   | 43.84                       | 40.96 | 56.35 | 15.73   | 26.55 | 29.86 | 32.37   | 40.59 | 34.06 |
|                    | PFMFCC | 51.06                       | 56.01 | 57.23 | 23.58   | 33.75 | 17.31 | 35.78   | 34.78 | 33.18 |
|                    | MODGD  | 46.15                       | 51.16 | 45.32 | 26.98   | 19.78 | 33.56 | 36.66   | 36.60 | 35.96 |
| Turkish Training   | MFCC   | 26.41                       | 30.70 | 33.92 | 47.44   | 34.16 | 47.56 | 34.75   | 30.70 | 27.26 |
|                    | LFCC   | 29.01                       | 28.82 | 33.36 | 42.31   | 37.46 | 28.76 | 31.64   | 27.14 | 33.24 |
|                    | RFCC   | 30.02                       | 26.32 | 36.61 | 47.19   | 41.65 | 49.42 | 29.50   | 26.90 | 34.48 |
|                    | PFMFCC | 30.31                       | 32.53 | 34.19 | 44.12   | 38.74 | 49.85 | 30.70   | 29.07 | 28.08 |
|                    | MODGD  | 28.27                       | 17.05 | 30.77 | 37.04   | 37.68 | 50.25 | 32.64   | 27.59 | 32.16 |
| English Training   | MFCC   | 32.16                       | 24.28 | 42.73 | 54.65   | 32.90 | 34.25 | 47.45   | 45.97 | 51.44 |
|                    | LFCC   | 35.96                       | 33.64 | 34.11 | 40.12   | 43.05 | 34.65 | 44.39   | 44.03 | 49.32 |
|                    | RFCC   | 30.49                       | 32.07 | 39.30 | 42.57   | 42.85 | 28.85 | 41.31   | 43.64 | 47.72 |
|                    | PFMFCC | 31.14                       | 27.90 | 34.11 | 43.21   | 38.55 | 34.33 | 45.88   | 46.99 | 50.68 |
|                    | MODGD  | 35.77                       | 33.46 | 36.15 | 34.09   | 29.71 | 27.93 | 34.03   | 35.06 | 33.73 |

For instance, it is quite strange to see 57.63% and 60.57% accuracies for MFCC and PFMFCC for PLDA model trained with English utterances of the female dataset tested with Turkish female datasets respectively. PLDA generally showed better performance.

The extent of performance degradation due to language mismatch can go as low as accuracy levels of 19.89% and 13.43% in the female and male datasets. These lowest accuracies are recorded for RFCC feature on Turkish trained GMM classifier tested

with English utterances and LFCC feature set on Turkish trained GMM classifier tested with German utterances respectively.

With the Turkish male dataset for instance, a maximum of 32.54%, 37.92% and 33.64% accuracy gains are recorded over a cross-language performance against German trained model using matched-language, multilingual and bilingual training setups respectively. These improvements are made with a feature-classifier pairs of PFMFCC-PLDA, RFCC-CDS and RFCC-CDS in their respective order. Likewise, 22.32%, 11.66% and 16.11% maximum accuracy improvements are made over cross-language evaluation of English trained MODGD-PLDA, MODGD-CDS and MODGD-PLDA feature-classifier models respectively. Table 6.7 presents a comparison of best performances in each 6 dataset for the matched-language training setup with Multilanguage and cross-language scenarios. In addition the performance evaluation of a bilingual setup is also included for German and Turkish databases. These figures should only be compared horizontally as they represent evaluation of the same feature-classifier pair along a certain row. However, if we look at vertically down for a given column the values may not be from the same model.

Table 6.7. Performance comparison of best matched-language classification accuracies with multilingual and cross-language scenarios

|                | Feature-classifier pairs | Training setups (accuracies in %) |              |           |                |        |         |
|----------------|--------------------------|-----------------------------------|--------------|-----------|----------------|--------|---------|
|                |                          |                                   |              |           | Cross-language |        |         |
|                |                          | Matched                           | Multilingual | Bilingual | English        | German | Turkish |
| Turkish male   | MODGD-PLDA               | 50.25                             | 45.07        | 44.04     | 33.56          | 27.93  |         |
| Turkish female | RFCC-GMM                 | 70.60                             | 70.13        | 70.93     | 26.43          | 40.84  |         |
| German male    | PFMFCC-PLDA              | 57.23                             | 29.84        | 37.81     | 34.11          |        | 34.19   |
| German female  | RFCC-GMM                 | 58.52                             | 49.04        | 57.30     | 24.00          |        | 33.57   |
| English male   | MFCC-PLDA                | 51.44                             | 42.13        | 32.70     |                | 24.13  | 27.26   |
| English female | PFMFCC-GMM               | 46.27                             | 45.92        | 25.60     |                | 32.90  | 22.73   |

The bilingual training setup is composed of utterances from German and Turkish databases. English test sets are applied to this scenario to see how the absence of a language can affect the performance. Indeed it showed 20.32% and 9.43% decline in performance compared to the multilingual setup with the three databases for female and male English datasets respectively. The PLDA classifier failed to make a positive contribution in the German male test set for bilingual and multilingual scenarios. It even performed worse than the cross-language setups. This could be due to the imbalance in the Turkish male dataset and differences in sequence of

phonemes in test and training samples which might have played a negative role the performance. However, some features made improvements with this classifier. For example

### 6.1.3. Regression results and speech duration analysis

LSSVR results are presented in Tables 6.7 and 6.8 for male and female datasets of the aGender database. Two scenarios are considered; LSSVR applied on i-vectors of 5 spectral feature sets and directly on the acoustic feature sets concatenated to form a supper vector. The unit  $f$  stands for frames which basically contain 20 milliseconds of speech.

Table 6.8. Performance evaluation in a) MAE b)  $\rho$  of feature+i-vector+LSSVR method for male dataset

a) Mean absolute error (MAE) for male dataset

| Male frames | PFMFCC       | MFCC         | RFCC         | LFCC         | MODGD        |
|-------------|--------------|--------------|--------------|--------------|--------------|
| 50f         | 7.598        | 7.842        | 7.997        | 8.272        | 8.189        |
| 100f        | 6.809        | 6.733        | 7.288        | 7.423        | 7.284        |
| 200f        | 6.340        | 6.231        | 7.131        | 6.969        | 6.776        |
| 400f        | 6.142        | 6.027        | 7.458        | 6.955        | 6.570        |
| 800f        | 6.130        | 6.022        | 7.072        | 6.925        | 6.515        |
| 1000f       | <b>6.129</b> | <b>6.015</b> | <b>7.046</b> | <b>6.924</b> | <b>6.501</b> |

b) Pearson correlation coefficient  $\rho$  for male dataset

| Male frames | PFMFCC       | MFCC         | RFCC         | LFCC         | MODGD        |
|-------------|--------------|--------------|--------------|--------------|--------------|
| 50f         | 0.601        | 0.578        | 0.558        | 0.528        | 0.543        |
| 100f        | 0.682        | 0.690        | 0.632        | 0.620        | 0.578        |
| 200f        | 0.724        | 0.731        | 0.650        | 0.666        | 0.679        |
| 400f        | <b>0.748</b> | 0.757        | 0.609        | 0.671        | 0.696        |
| 800f        | 0.747        | <b>0.758</b> | <b>0.652</b> | 0.674        | 0.701        |
| 1000f       | 0.746        | 0.746        | 0.654        | <b>0.674</b> | <b>0.702</b> |

Table 6.9. Performance evaluation in a) MAE b)  $\rho$  of feature+i-vector+LSSVR method for female dataset

a) Mean absolute error (MAE) for female dataset

| Female frames | PFMFCC       | MFCC         | RFCC         | LFCC         | MODGD        |
|---------------|--------------|--------------|--------------|--------------|--------------|
| 50f           | 8.004        | 8.369        | 8.643        | 8.520        | 8.535        |
| 100f          | 7.103        | 7.615        | 7.415        | 7.462        | 7.297        |
| 200f          | 6.633        | 7.222        | 6.731        | 6.804        | 6.862        |
| 400f          | 6.411        | 6.836        | 6.348        | 6.547        | 6.610        |
| 800f          | 6.363        | 6.825        | 6.247        | 6.460        | 6.604        |
| 1000f         | <b>6.363</b> | <b>6.816</b> | <b>6.219</b> | <b>6.457</b> | <b>6.583</b> |

b) Pearson correlation coefficient  $\rho$  for female dataset

| Female frames | PFMFCC       | MFCC         | RFCC         | LFCC         | MODGD        |
|---------------|--------------|--------------|--------------|--------------|--------------|
| 50f           | 0.653        | 0.619        | 0.601        | 0.618        | 0.603        |
| 100f          | 0.721        | 0.674        | 0.701        | 0.700        | 0.698        |
| 200f          | 0.763        | 0.722        | 0.765        | 0.771        | 0.736        |
| 400f          | 0.779        | 0.753        | 0.790        | 0.791        | 0.752        |
| 800f          | 0.782        | 0.753        | 0.797        | 0.796        | 0.753        |
| 1000f         | <b>0.782</b> | <b>0.753</b> | <b>0.799</b> | <b>0.797</b> | <b>0.755</b> |

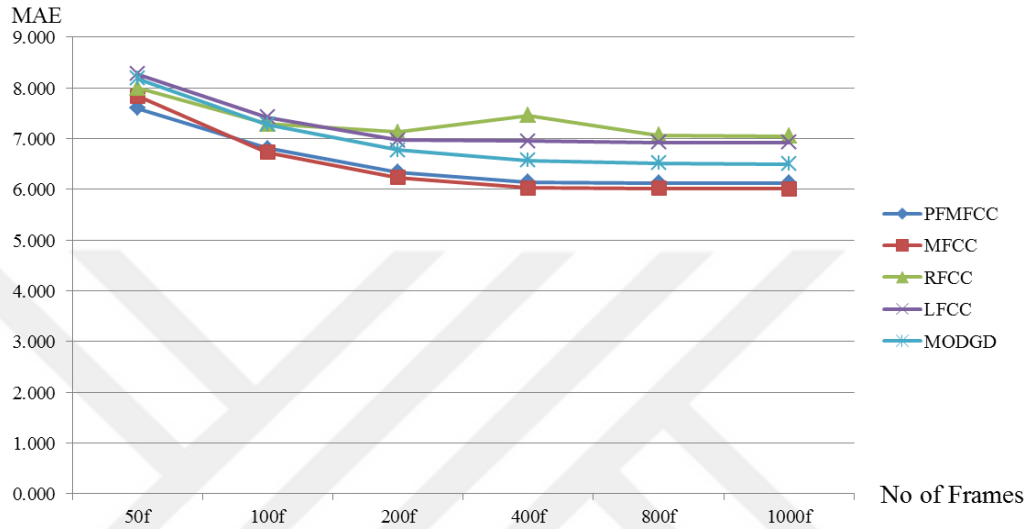


Figure 6.3. MAE of LSSVR expressed along increasing number of frames for male aGender dataset

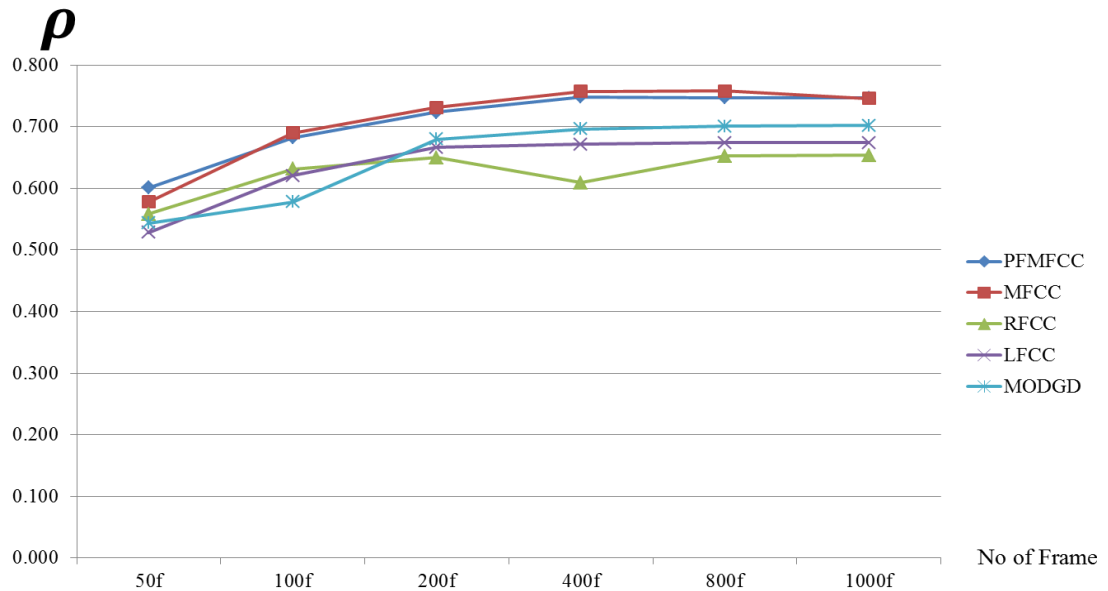


Figure 6.4.  $\rho$  as frames increase for LSSVR for male aGender dataset

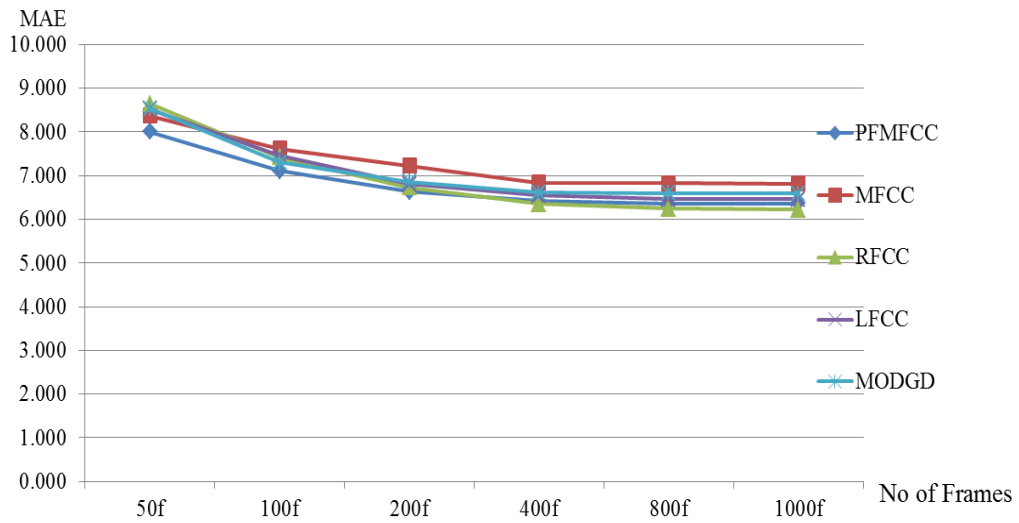


Figure 6.5. MAE of LSSVR expressed along increasing number of frames for female aGender dataset

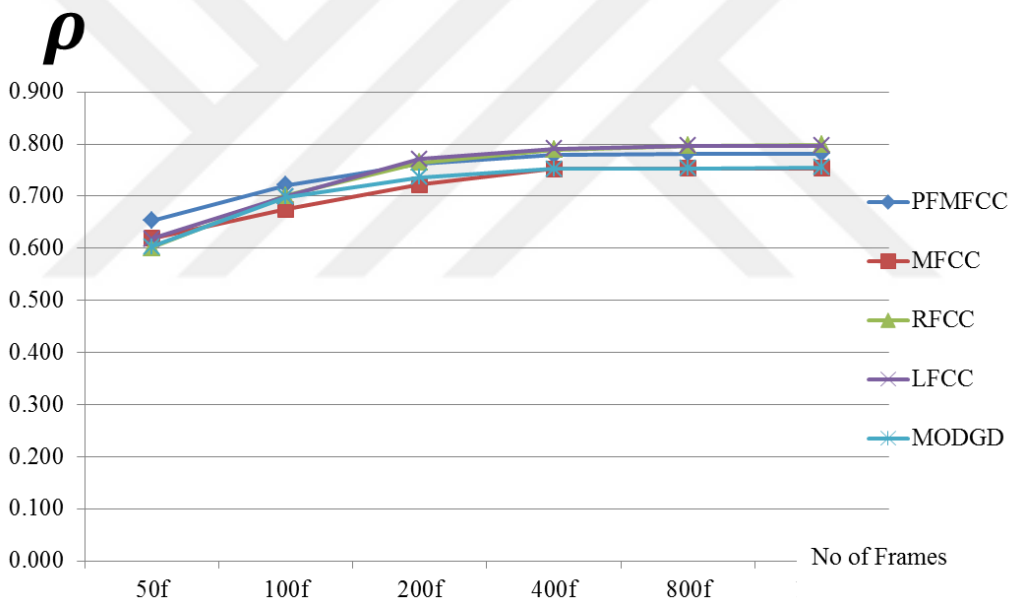


Figure 6.6.  $\rho$  as frames increase for LSSVR over female aGender dataset

In addition, Figures 6.3 and 6.4 above present the MAE and  $\rho$  graphically for different speech durations expressed in terms of number of frames for the male dataset. Figures 6.5 and 6.6 present for the female dataset likewise.

Tables 6.9 and 6.10 below, show the performance evaluation of the LSSVR regression algorithm on i-vector sequences for different combination of mismatch in length of utterances in speech segments assuming 200, 500 and 1000 frames as short

medium and long utterances respectively. The values located across the diagonals represent matched and the bold values show best mismatch performances.

Table 6.10. i-Vector followed by LSSVR performance evaluation for utterance length mismatch in terms of MAE for female dataset. (Rows are training and columns are test frames)

|                     |      | a) PFMFCC-i-vector-LSSVR |        |               | b) MFCC-i-vector-LSSVR |        |               |
|---------------------|------|--------------------------|--------|---------------|------------------------|--------|---------------|
|                     |      | Test frames              |        |               | Test frames            |        |               |
|                     |      | 200                      | 500    | 1000          | 200                    | 500    | 1000          |
| Training frame size | 200  | 6.6330                   | 6.4602 | 6.4346        | 7.2220                 | 6.8266 | 6.7969        |
|                     | 500  | 6.7843                   | 6.4467 | <b>6.4255</b> | 7.2520                 | 6.8224 | <b>6.7935</b> |
|                     | 1000 | 6.8052                   | 6.4717 | 6.3630        | 7.2793                 | 6.8467 | 6.8160        |
|                     |      | c) RFCC-i-vector-LSSVR   |        |               | d) LFCC-i-vector-LSSVR |        |               |
|                     |      | Test frames              |        |               | Test frames            |        |               |
|                     |      | 200                      | 500    | 1000          | 200                    | 500    | 1000          |
| Training frame size | 200  | 6.7310                   | 6.3952 | 6.3650        | 6.8040                 | 6.4187 | 6.3945        |
|                     | 500  | 6.6540                   | 6.2619 | <b>6.2190</b> | 6.6744                 | 6.3622 | <b>6.3350</b> |
|                     | 1000 | 6.6500                   | 6.2607 | 6.2190        | 6.6514                 | 6.3380 | 6.4570        |

Table 6.11. i-Vector followed by LSSVR performance evaluation for utterance length mismatch in terms of MAE for male dataset. (Rows are training and columns are test frames)

|                     |      | a) PFMFCC-i-vector-LSSVR |               |               | b) MFCC-i-vector-LSSVR |               |        |
|---------------------|------|--------------------------|---------------|---------------|------------------------|---------------|--------|
|                     |      | Test frames              |               |               | Test frames            |               |        |
|                     |      | 200                      | 500           | 1000          | 200                    | 500           | 1000   |
| Training frame size | 200  | 6.3400                   | 6.1726        | 6.7498        | 6.2309                 | 6.1849        | 6.1544 |
|                     | 500  | 6.4308                   | 6.1736        | 6.1982        | 6.1500                 | 6.0680        | 6.0424 |
|                     | 1000 | 6.3770                   | <b>6.1555</b> | 6.1285        | 6.1354                 | <b>6.0387</b> | 6.0147 |
|                     |      | c) RFCC-i-vector-LSSVR   |               |               | d) LFCC-i-vector-LSSVR |               |        |
|                     |      | Test frames              |               |               | Test frames            |               |        |
|                     |      | 200                      | 500           | 1000          | 200                    | 500           | 1000   |
| Training frame size | 200  | 7.1306                   | 7.0903        | 7.0717        | 6.9685                 | 6.9723        | 6.9873 |
|                     | 500  | 7.1092                   | 7.0687        | <b>7.0480</b> | 6.9442                 | 6.9328        | 6.9380 |
|                     | 1000 | 7.0919                   | 7.0715        | 7.0459        | 6.9337                 | <b>6.9219</b> | 6.9243 |

Performance comparison of LSSVR model on direct acoustic spectral features and i-vectors as a second tier feature extraction for utterance lengths of 3, 5 and 10 seconds as short medium and long speech utterances respectively is made and presented in Table 6.11 below. The input sequences used in acoustic feature sets with LSSVR regression model are extremely long as they are a result of concatenation of several frames. Hence the performance is not only poor but also slow due to the length of sequences.



The LSSVR is not only ineffective, but also slow when applied to the super-vector acoustic feature sets directly before i-vectors are generated from them as the concatenation of all the frames in the entire speech segment makes the dimension of observations extremely large. If we look at the shortest speech segment, i.e. 3 seconds containing 300 frames for instance, each frame consists of 42 feature values which would make up a super-vector of dimension  $d = 300 \times 42 = 12600$ .

Table 6.12. Performance of LSSVR model on short, medium and long utterances for female and male datasets

| Duration                                       | Female MAE/ $\rho$ | Feature set used and improvement | Male MAE/ $\rho$ | Feature set used and improvement |
|------------------------------------------------|--------------------|----------------------------------|------------------|----------------------------------|
| <b>3s</b>                                      |                    |                                  |                  |                                  |
| Feature + LSSVR                                | 11.704/0.580       | RFCC, 44.98% improvement         | 11.093/ 0.500    | MFCC, 44.77% improvement         |
| Feat +i-vector + LSSVR                         | 6.439/0.781        |                                  | 6.127/0.746      |                                  |
| <b>5s</b>                                      |                    |                                  |                  |                                  |
| Feature + LSSVR                                | 11.628/0.592       | RFCC, 46.15% improvement         | 11.063/0.504     | MFCC, 45.15% improvement         |
| Feat +i-vector + LSSVR                         | 6.262/0.796        |                                  | 6.068/0.7526     |                                  |
| <b>10s</b>                                     |                    |                                  |                  |                                  |
| Feature + LSSVR                                | 11.555/0.594       | RFCC, 46.179% improvement        | 11.012/ 0.506    | MFCC, 45.38% improvement         |
| Feat +i-vector + LSSVR                         | 6.219/0.799        |                                  | 6.015/0.746      |                                  |
| Note: Feat = {MFCC, RFCC, LFCC, PFMFCC, MODGD} |                    |                                  |                  |                                  |

The complexity of regression increases and the speed of operation dramatically decreases compared to the regression applied on a fixed 200 i-vectors. The order of complexity is crucial especially when we are working with a large data. We need to consider every possibility to reduce the hurdles on our computing machines. In this regard the i-vector LSSVR approach has made 98.4% dimensionality reduction cutting the 12600 long, vector to only 200 identity vectors.

#### 6.1.4. Performance evaluation of deep learning based classifiers

Performance evaluation of speaker age classification for both the Turkish and aGender databases using x-vector neural network architecture with PLDA classifier

is presented in Table 6.12 below. Our model does not perform well for the male dataset in the Turkish database compared to other datasets due to imbalance in number of utterances in each class. Senior speakers are not represented sufficiently in this dataset.

Table 6.13. Cross-gender speaker age evaluation using x-vector neural network architecture

| German |        | Test   |               |
|--------|--------|--------|---------------|
|        |        | Male   | female        |
| Train  | male   | 54.588 | 35.349        |
|        | female | 36.372 | <b>57.565</b> |

| Turkish |        | Test   |               |
|---------|--------|--------|---------------|
|         |        | male   | female        |
| Train   | male   | 44.709 | 32.828        |
|         | female | 33.368 | <b>64.687</b> |

The x-vector neural network is tested with utterances drawn from an unseen and unrepresented datasets. The results shown in Table 6.13 are comparable to cross-language and multi-language performance evaluation of GMM, SVM and feedforward DNN carried out for Turkish and German speech utterances in a previous literature [17]. However, it has been observed that a significant increase in accuracies was unexpectedly made by different gender evaluations for aGender training and Turkish test sets in our experiment.

Table 6.14. Cross-language and cross-gender speaker age evaluation using x-vector neural network architecture

| Turkish-German |        | German test set |               |
|----------------|--------|-----------------|---------------|
|                |        | Male            | Female        |
| Turkish Train  | male   | 31.510          | 31.814        |
|                | female | 34.419          | <b>40.696</b> |

| German-Turkish |        | Turkish test set |               |
|----------------|--------|------------------|---------------|
|                |        | Male             | Female        |
| German Train   | male   | 29.935           | <b>48.190</b> |
|                | female | <b>41.565</b>    | 36.000        |

Performance evaluation of 5 classifiers which include both classical and deep neural network (DNN) based models on the aGender and Turkish databases are summarized in Table 6.14 below. The classical machine learning models include GMM, CDS and PLDA. On the other hand the remaining two DNN based classifiers are LSTMM and x-vector neural network architecture. LSTM offered accuracies of 51% and 64.88% for English and Turkish female datasets respectively with MFCC features as an input. A cross-language evaluation on this classifier resulted 41.28% and 35.39% for Turkish and German female datasets with the English training setup respectively. Likewise, cross-language evaluation on the Turkish trained model offered 35.6% and 33.31% for German and English female datasets respectively. An end-to-end experimental setup with the LSTM classifier offered an accuracy of 58.61% which is the highest compared to all other performances on the female dataset of the aGender database.

Table 6.15. Performance evaluation of 5 classifiers with MFCC sequences for speaker age classification on a) female b) male datasets respectively

|             |          | Accuracies in % |       |         |       |
|-------------|----------|-----------------|-------|---------|-------|
|             |          | aGender         |       | Turkish |       |
|             |          | Female          | Male  | Female  | Male  |
| Classifiers | CDS      | 56.44           | 41.80 | 62.24   | 47.44 |
|             | GMM      | 57.03           | 42.63 | 70.45   | 34.16 |
|             | PLDA     | 57.03           | 53.75 | 51.98   | 47.56 |
|             | LSTM     | 56.69           | 47.73 | 64.88   | 43.54 |
|             | x-Vector | 57.57           | 54.59 | 64.69   | 44.71 |

## 6.2. Discussion

A total of ten feature extraction methods are implemented in our experiments. We observed their variation across selected classifiers and regression models. An end-to-end classification is also carried out with the LSTM deep neural network classifier where the feature extraction stage is ignored and offered the best result in the female dataset of aGender database. We selected some relevant experimental results and presented their interpretation here in this sub section. We conducted the experiments a single language (matched-language), bilingual, multilingual and cross-language scenarios.

### 6.2.1. PFMFCC versus MFCC for speaker age classification

The proposed feature set PFMFCC resembles the popular MFCC spectral feature in its algorithm. It only replaces the triangular band pass filters with parabolic shapes. The sharp corners at the top of each triangular filter bank in MFCC would make it hard to imagine practical implementation of these shapes. On the contrary, practical low pass, or band pass filters can approximate the parabolic shaped bank of filters to generate PFMFCC feature sets.

After critically examining most research studies carried out on speaker age classification we found out that they heavily depend on MFCC features. Their focus is on the backend (classifiers). Inspired by the performance of some features for speech recognition, speaker recognition, speaker emotion recognition, speaker diarisation (diarization) and replay attack detection; we inquired if these features could perform well on speaker age classification and decided to apply them in our study. Following how the filter band based features are extracted cautiously, we developed more efficient, effective and practical algorithm to generate a new set of features using parabolic band pass filter banks. Table 6.16 below presents the summary of comparisons between the popular MFCC with our proposed feature set PFMFCC. Based on this summary we can say that PFMFCC contains more age information than MFCC.

Table 6.16. MFCC versus PFMFCC

| Criteria                             | MFCC                                                                | PFMFCC                                                              |
|--------------------------------------|---------------------------------------------------------------------|---------------------------------------------------------------------|
| Filter bank shape                    | Triangular                                                          | Parabolic                                                           |
| No of functions used per filter bank | 2 linear functions                                                  | A single polynomial function of degree 2                            |
| Number of features in a frame        | 13 static + 13 dynamic + 13 acceleration + 3 Energy components = 42 | 13 static + 13 dynamic + 13 acceleration + 3 Energy components = 42 |
| Performance for female dataset       | Cosine score 56.44%, GMM 57.03%, PLDA 57.03%                        | Cosine score 51.06%, GMM 56.01%, PLDA 58.14%                        |
| Performance for male dataset         | Cosine score 41.8%, GMM 42.63%, PLDA 53.75%                         | Cosine score 50.44%, GMM 52.74%, PLDA 57.23%                        |
| Realizability                        | Not easier to implement as it consists of a sharp corner.           | Can be approximated with practical filters.                         |

### 6.2.2. Unexpected effect of VAD

Voice activity detection (VAD) also known as speech activity detection (SAD), reduces the amount of data by removing non-speech frames from utterances [123]. It usually improves performance in speech recognition, speech coding and other linguistic based speech processing applications. However, speaker age is a paralinguistic attribute which also depends on non-verbal contents such as tone and pitch unlike the former applications that highly rely on linguistic content of speech. Therefore, the benefit of VAD seems to be insignificant in this regard or energy parameters need to be re-adjusted in order to meet its goal. Surprisingly, VAD degraded the performance of age classification in the PLDA classifier noticeably. This raises skepticism. But it could largely be due to double reductions when VAD is applied before PLDA scheme. The first one is reduction in the amount of data due to removal of non-speech frames mainly consisting silence and noise. The second one is dimensionality reduction using LDA right before PLDA scoring.

The frames removed by VAD may have contained important patterns related to age as the energy threshold is not well crafted to preserve age related information in low energy frames. Therefore, we suggest different VAD parameters for different speech processing applications. Figure 6.7 depicts VAD and non VAD scenarios for speaker age classification using PLDA for both male and female datasets of the aGender database [76].

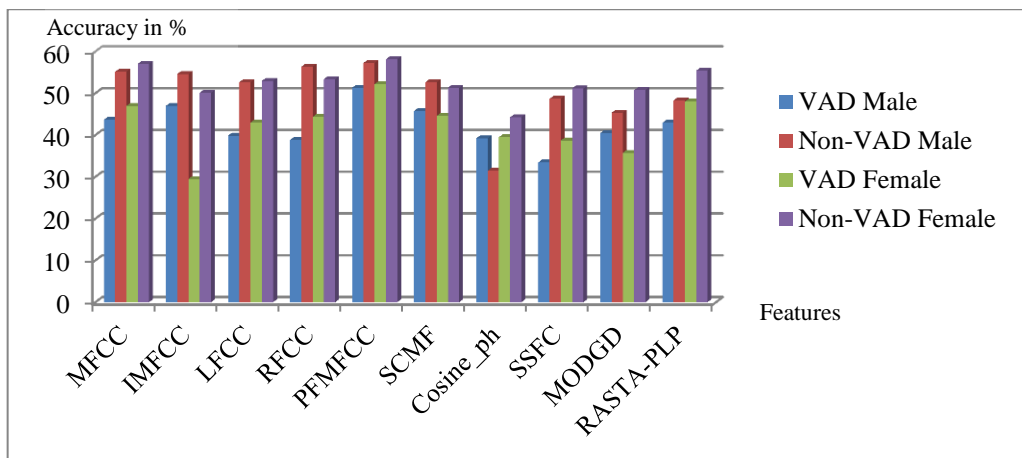


Figure 6.7. Effect of VAD on the PLDA classifier for male and female datasets of aGender database from simulation results

Using a different energy threshold has improved the performance of PLDA classifier with MFCC and PFMFCC feature sets in the Turkish male dataset by 8.81% and 7.40% respectively. Two criteria are used to remove the silent and noise only frames. Either frames with energy below -60dB or if the maximum energy among all the frames in an utterance is above -20dB a relative criteria is applied and frames with energy below 40dB below the maximum energy will be removed. To make the second criteria more clear, let us assume we have 200 frames in a certain utterance. After computing the energy of each frame we found out the maximum energy among the 200 frames is -30dB. Hence, we use the first criteria (absolute criteria) because the maximum energy is below -20dB. Accordingly, those frames with energy below -60dB will be discarded. However, if the maximum energy was -10dB instead, the second criteria (relative criteria) would be applied as -10dB is above -20dB. Therefore, those frames with energy below (-10-40=-50dB) would be discarded.

The performance variations especially in the PLDA classifier for aGender and Turkish datasets, indicates that the noise characteristics in the two databases has contributed either positively or negatively. In the Turkish database the noise characteristics affected the performance negatively as it is proved above. On the other hand, it has played a positive role in the performance improvement of the PLDA classifier for the German database. Obviously, the noise characteristics in the two databases, is different as the recording is done through telephone line and directly through computer for the German and Turkish databases respectively.

On the other hand, GMM and CDS did not show significant difference in performance with VAD and without VAD scenarios. Hence we can make a tradeoff either to remove non-speech frames from all utterances or ignore it and make the training processes busy with less relevant noise and silence frames in these two classifiers. The choice is clear; we have to remove these frames as the training step is a long process compared to VAD, it would definitely make a difference in improving speed.

### **6.2.3. Performance of feature fusion**

In addition to standalone performance selection of feature sets based on their individual performance is made and feature fusion is carried out on different kinds of

features. At first a fusion of all the ten features was done using concatenation which resulted in a performance below the best performance of a single feature set. In the next steps few feature sets that showed the worst performances were removed one feature set at a time and arrived at a fusion of seven feature sets that consists of MFCC, MODGD, RFCC, SCMF, SSFC, RASTA-PLP and PFMFCC. VAD is not applied on these features. The CDS classifier gave the best performance in both genders with 62.14% and 59.54% accuracies for female and male datasets respectively as shown in Figure 6.8 below.

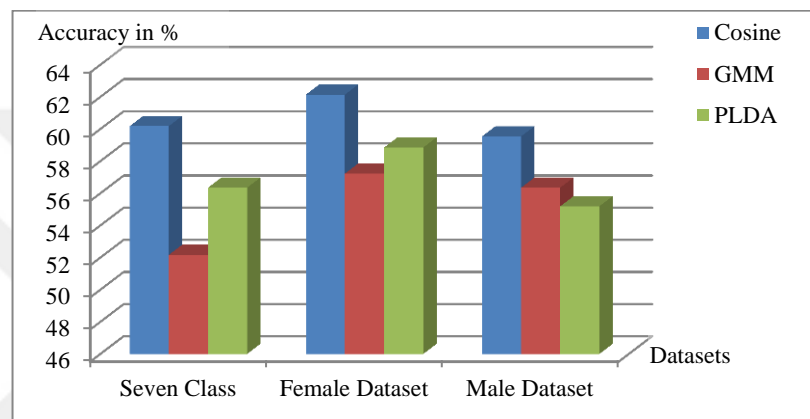


Figure 6.8 Performance evaluation results of feature fusion of seven feature sets on three classifiers on seven and three age class arrangements

The Matlab simulation result carried out on all the seven classes consisting children aged 7-14, young female aged 15-24, adult female aged 25-54, old female aged 55-80, young male aged 15-24, adult male aged 25-54 and old male aged 55-80 resulted in overall accuracies of 60.18%, 52.17% and 56.35% using CDS, GMM and PLDA respectively. The age classes are made based on the aGender database [20]. According to this result the cosine score classifier has made an overall improvement of the accuracy by 2.55% compared to speaker age classification study carried out on the same database in [119].

#### 6.2.4. Limitations, solutions and findings

An apparent limitation of our study lays on the difficulty of finding convenient boundaries especially between adult and old speaker age classes where one speaker age class ends and the next one begins. For instance putting age 55 together with age

80 and putting age 54 in a different speaker age class causes a great deal of error. This problem could be solved by using regression instead of classification with a large amount of training data which can represent each and every age sufficiently. With a large amount of data and sufficient representation of discrete ages, regression can avoid the inconvenience of age class boundaries. However, we also suggest binary classification of adjacent age bins of selected ages around the boundaries before the actual speaker age classification experiment to find suitable boundaries.

Another limitation of our study is that the databases especially the aGender consist of short utterances (2.55 second on average) which also include non-speech frames. VAD further reduces the number of frames in an utterance by discarding the non-speech frames. Non-speech frames contain spectral energy below a certain threshold. Furthermore, PLDA reduces dimension due to its linear discriminant analysis (LDA) function within it. The performances of features with and without VAD on PLDA classifier as shown in Figure 6.7 indicates that PFMFCC and fusion of features could perform better on a database with longer duration of utterances.

The imbalance in the size of each class in the Turkish male dataset caused significant decline in performance compared to other databases. Addition of the German database for a bilingual training scenario improved the performance of the PLDA classifier. Additionally, the CDS classifier outperformed with a multilingual scenario over matched language setup for certain feature sets such as MFCC, LFCC and RFCC. Therefore, it is likely that these features overcome the imbalance problem in a single language setup with addition of more languages to the training. It can also be imagined that the CDS model in multilingual and PLDA model in bilingual scenarios are able to learn age classes from the phoneme sequences of heterogeneous language scenarios better compared to a monolingual classification over this dataset.

The sampling rate in the three databases is different, for this reason down sampling was required in order to carry out bilingual and multilingual trainings. As the aGender database is most complete and balanced as well as recorded mainly with the intension of speaker age classification the age classes and the sampling rates in the Turkish and Age-Vox-Celeb databases are brought to be similar with aGender.



Apart from few outliers which have been challenging for interpretation, multilingual and bilingual scenarios have improved the performance significantly compared to cross-language setups. To mention few strange results in the cross-language evaluation; 60.56% accurate classification of female Turkish test sets by English trained PLDA classifier using PFMFCC feature sets as an input which is more than 20% better than the matched-language performance with the same classifier and feature, 54.56% accurate age classification of Turkish male test audios by English trained CDS classifier using MFCC feature sets which showed 7.21% increase compared to a Turkish trained CDS classifier performance with the same feature, and the MODGD features offered better performances with all the three classifiers over German-Turkish cross-language scenario compared to Turkish-Turkish matched language setting. This could luckily be due to the phoneme sequences in the test sets which may have enabled the models to learn better than matched-language scenarios.

Nevertheless, the best performances for the matched-language scenario in each datasets are in harmony with our expectations compared to other scenarios. The summary of best performances by matched-language scenarios for each of the three databases and both genders is presented in Table 6.7. These results are compared to multilingual, bilingual and cross-language scenarios for the same dataset as well as feature-classifier pair and presented in Figure 6.9 for better visualization. If we look at same efforts over these datasets especially the German and the Turkish; our results showed 5.6% increase in accuracy with MFCC using the PLDA classifier for the Turkish male dataset exploiting bilingual training compared to SVM classifier with MFCC carried out in [17] which offered 50% accuracy for the same dataset and feature. The MODGD and PFMFCC features also showed 1.48% and 1.13% with the same classifier and dataset over SVM respectively. Similarly, accuracies of 54.5%, 69.2% and 69.6% are reported with GMM, SVM and DNN classifiers respectively with a multi-language training of German and Turkish female audios in [17]. On the other hand our GMM classification experiment on the same multilingual dataset resulted in accuracies of 70.16%, 70.45% and 70.93% with MFCC, LFCC and RFCC features respectively. Hence our best result showed an improvement by 1.33% over the DNN classification. Our approaches also outperformed in the German female

dataset which gained 17.19%, 7.6% and 9.9% accuracy increases with RFCC feature and GMM classifier over GMM, SVM and DNN respectively in [17].

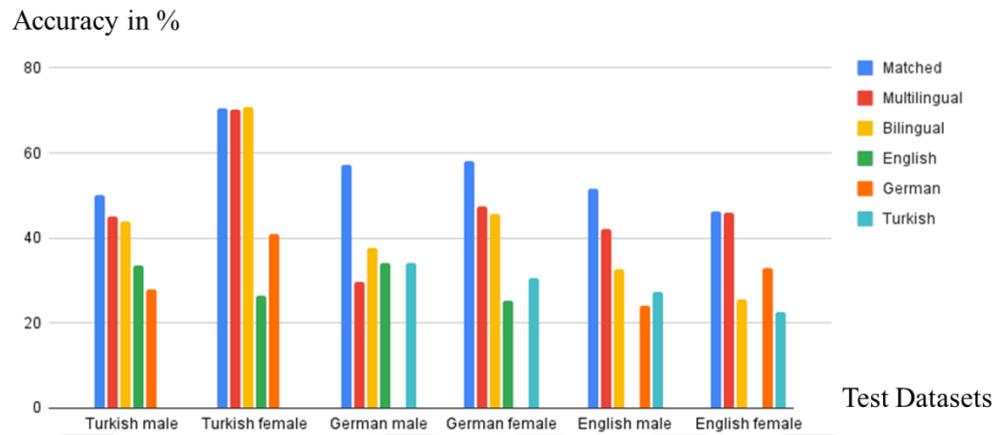


Figure 6.9. Performance comparison of matched-language, multilingual, bilingual and cross-language training scenarios for speaker age classification

## 7. CONCLUSION

Speaker age prediction has remained one of the most challenging research disciplines in speech processing due to the stochastic nature of speech signals despite all the concerted efforts to improve age prediction metrics. With the reemergence of DNNs, innovation of high speed digital signal processors (DSPs) and intelligent machines this area of research has been getting more interest in recent years. A breakdown of the difficulties has been made and some solutions have been proposed to resolve certain challenges in this research. Additional algorithms and techniques have been introduced in order to fill the research gap in this area. PFMFCC has been designed and proposed as an alternative to existing spectral feature extraction techniques. Several feature sets, which have not been applied to speaker age prediction before, are adopted to this research and observed to make a difference with certain classifiers. This study has also strengthened the formation of multilingual training scenarios which has remained rare in speaker age estimation.

This study investigated certain factors affecting speaker age prediction. The predictions were implemented in the form of classification and regression. Spoken language, amount of training data, speech duration, asymmetry of utterances representing age classes and environmental noise were the presupposed determinants confirmed to alter speaker age prediction.

Speech duration is confirmed to affect the prediction of speaker age in this study. It is in line with the hypothesis we presupposed that longer utterances deliver better performances than shorter ones. Speaker age regression using LSSVR experimental results confirmed this argument that the improvement continued from 0.5 to 4 seconds to be bold and started to saturate all the way to 10 second irrespective of front end feature set type. The improvement continued after 4 second but remained too little to bargain over computation overheads due to large data size. Therefore, developers are recommended to train their prediction models with medium length

utterances typically 4-6 seconds which is capable of tolerating utterance length mismatches during performance evaluation.

Several magnitude, phase and sub-channel based spectral features are employed for speaker age estimation with classification given more emphasis in this research. Our proposed feature set PFMFCC overwhelmingly outperformed the majority of the remaining features and offered relatively close results with the famous MFCC feature on all classifiers and regression models. With the PLDA classifier it gave the best performance for both genders and all database setups.

Despite the high order of complexity exhibited during simulations, LSTM offers best speaker classification and regression performances for female and male datasets of the aGender database with power spectral sequences with no further processing at the frontend. PLDA offered comparatively best results among traditional machine learning models for speaker age classification.

Multilingual training has been observed to make up for the poor performance of classification models due to language mismatches between training and test datasets. The performance for matched language setups is observed to be the best among the three scenarios; matched-language, language mismatch (cross-language) and multilingual setups. The multilingual training however, does not affect the performance of a matched-language significantly while it played a crucial role in improving the prediction accuracy for cross-language settings. Increasing the number of languages in the multilingual scenario has even improved some of the feature-classifier pair performances further for certain datasets. The worst case scenario appears when the setup is made to be cross-language where the performance dropped dramatically. The unbalanced nature of the Turkish male dataset seems to cause performance decline with the German male test set during bilingual training scenarios. However, this training has offered benefits to the Turkish male evaluation significantly. Despite few outliers both the bilingual and multilingual (with the three languages) scenarios outperformed the cross-language evaluation in many of feature-classifier pairs. Therefore, multilingual training is preferred in order to widen the domain of incoming test set languages and get relatively acceptable accuracy of prediction.

Most speech processing applications heavily rely on magnitude based spectral feature sets. The impressive speaker age classification accuracies shown by the MODGD feature however, indicate age information is lumped in both magnitude and phase components of speech spectrum. Therefore, we recommend a combined strategy could outperform existing standalone approaches. The classification accuracies are as bad as chance level predictions for mismatch scenarios. Hence for its versatility, training a certain model with multilingual data is recommended.

The significant accuracy difference in the VAD and non-VAD experimental setups observed in PLDA classifier, suggests that a different energy threshold is required for different speech processing applications while removing non speech and noise frames from utterances during feature extraction. The nature of our databases may have caused the performance degradation of the PLDA classifier in VAD applied scenarios. Some frames are not sufficient enough to represent speech sound but, they may contain attributes that can enable speaker age recognition. Therefore, a lower energy threshold is recommended to retain some non-speech but, paralinguistic contents in speaker age estimation.

## REFERENCES

- [1] Schuller B., Steidl S., Batliner A., Burkhardt F., Devillers L., MüLLer C., Narayanan S., Paralinguistics in Speech and Language—State-of-the-art and the Challenge, *Computer Speech & Language*, 2013, **27**(1), 4–39.
- [2] Schultz T., Kirchhoff K., *Multilingual Speech Processing*, 1st ed., Elsevier, 2006.
- [3] Feld M., Barnard E., Van Heerden C., Müller C. Multilingual speaker age recognition: Regression analyses on the Lwazi corpus, in *IEEE Workshop on Automatic Speech Recognition Understanding*, Moreno, Italy, 13 Nov.-17 Dec 2009.
- [4] Boeree C. G., Abraham Maslow: *Personality theories*, 1–11, 2006.
- [5] Barbieri M., On the origin of language, *Biosemiotics*, 2010, **3**(2), 201–223.
- [6] DNA blood test predicts suspects' age, <http://news.yahoo.com/dna-blood-test-predicts-suspects-age-115435458.html> (Accessed date: Jul. 25, 2021).
- [7] Shamma M. A., Telomeres, lifestyle, cancer, and aging, *Current opinion in clinical nutrition and metabolic care*, 2011, **14**(1), 28–34.
- [8] Gałka J., Grzybowska J., Igras M., Jaciów P., Wajda K., Witkowski M., Ziółko M., System Supporting Speaker Identification in Emergency Call Center, In *Sixteenth annual conference of the international speech communication association*. Dresden, Germany 6 - 10 Sep. 2015.
- [9] Ortega-Garcia J., Gonzalez-Rodriguez J., Marrero-Aguilar V., AHUMADA: A large speech corpus in Spanish for speaker characterization and identification, *Speech communication*, 2000, **31**(2-3), 255-264.
- [10] Barkana B. D., Zhou J., A new pitch-range based feature set for a speaker's age and gender classification, *Applied Acoustics*, 2015, **98**, 52-61.
- [11] Li M., Jung C. S., Han K. J., Combining Five Acoustic Level Modeling Methods for Automatic Speaker Age and Gender Recognition, *Eleventh Annual Conference of the International Speech Communication Association Makuhari*, Japan, 26-30 Sept. 2010.
- [12] Dobry G., Hecht R. M., Avigal M., Zigel Y., Supervector Dimension Reduction for Efficient Speaker Age Estimation Based on the Acoustic Speech Signal, *IEEE Transactions on Audio, Speech, and Language Processing*, 2011, **19**(7), 1975–1985.

- [13] Hermansky H., Morgan, N., RASTA processing of speech, *IEEE Transactions on Speech and Audio Processing*, 1994, **2**(4), 578–589.
- [14] Schuller B., Steidl, S., Batliner A., Burkhardt F., Devillers L., Müller C., Narayanan S., The INTERSPEECH 2010 Paralinguistic Challenge, *Eleventh Annual Conference of the International Speech Communication Association*, Makuhari, Japan, 26-30 Sept. 2010.
- [15] Grzybowska J., Kacprzak S., Speaker Age Classification and Regression Using i-Vectors, *INTERSPEECH*, San Francisco, USA, 8-12 Sep. 2016.
- [16] Büyük O., Arslan M. L., Combination of Long-Term and Short-Term Features for Age Identification from Voice, *Advances in Electrical and Computer Engineering*, 2018, **18**(2), 101-108.
- [17] Büyük O., Arslan L. M., An Investigation of Multi-Language Age Classification from Voice, *BIOSIGNALS*, Prague, Czech Republic, 22-24 Feb. 2019.
- [18] Bahari M. H., Van Hamme H., Speaker age estimation and gender detection based on supervised Non-Negative Matrix Factorization, in *2011 IEEE Workshop on Biometric Measurements and Systems for Security and Medical Applications (BIOMS)*, Milan, Italy, 28 Sep. 2011.
- [19] Haniççi C., Features and classifiers for replay spoofing attack detection, *10th International Conference on Electrical and Electronics Engineering (ELECO)*, Bursa, Turkey, 30 Nov - 2 Dec. 2017.
- [20] Burkhardt F., Eckert M., Johannsen W., Stegmann J., A Database of Age and Gender Annotated Telephone Speech, *LREC*. Valletta, Malta, 17-23 May 2010.
- [21] Tawara N., Ogawa A., Kitagishi Y., Kamiyama H., Age-VOX-Celeb: Multi-Modal Corpus for Facial and Speech Estimation, in *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Toronto, Ontario, Canada, 6-11 June 2021.
- [22] Yang H. F., Lin B. Y., Chang K. Y., Chen C. S., Automatic age estimation from face images via deep ranking. *networks*, 2013, **30w5**(8), 1872-1886.
- [23] Higgs, P., Gilleard, C., The ideology of ageism versus the social imaginary of the fourth age: two differing approaches to the negative contexts of old age, *Ageing & Society*, 2020, **40**(8), 1617–1630.
- [24] ÖZKAN, Y., SERPEN, A. S., Ageism: College Students' Perceptions about Older People, *Social Sciences*, 2011, **6**(1), 107-115.
- [25] Myers West, S. Censored, suspended, shadowbanned: User interpretations of content moderation on social media platforms, *New Media & Society*, 2018, **20**(11), 4366–4383.

- [26] Howard, J. W., Free Speech and Hate Speech, *Annual Review of Political Science*, 2019, **22**(1), 93–109.
- [27] Safavi, S., Russell, M., Jančovič, P., Automatic speaker, age-group and gender identification from children’s speech, *Computer Speech & Language*, Jul. 2018, **50**, 141–156.
- [28] Whiteside S. P., Hodgson C., The Development of Fundamental Frequency in 6-to 10-Year Old Children: A Brief Study, *Journal of the International Phonetic Association*, 1998, **28**(1-2), 55–62.
- [29] Chollet F., *Deep learning with Python*. Shelter Island, NY: Manning, 2018.
- [30] Sadjadi S. O., *MSR Identity Toolbox*, Seattle, WA, USA: Microsoft, 2013.
- [31] Pépiot E., Male and female speech: a study of mean f<sub>0</sub>, f<sub>0</sub> range, phonation type and speech rate in Parisian French and American English speakers, *Speech Prosody*, 2014, **7**, 305-309.
- [32] Geng X., Zhou Z., Smith-Miles K., Automatic Age Estimation Based on Facial Aging Patterns, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2007, **29**(12), 2234–2240.
- [33] Schötz S., Acoustic Analysis of Adult Speaker Age, *Speaker classification I*. Springer, Berlin, Heidelberg, 2007. 88-107
- [34] Browman C. P., Goldstein L. Dynamics and articulatory phonology. *Mind as motion*, The MIT Press, 175-193, 1995.
- [35] Yoshida M., The Articulatory System,. [http://ocw.uci.edu/upload/files/the\\_articulatory\\_system.pdf](http://ocw.uci.edu/upload/files/the_articulatory_system.pdf) (Accessed date: Jan. 25, 2021).
- [36] Rabiner L. R., Schafer R. W., *Digital processing of speech signals*, Prentice Hall, New Jersey, 1978.
- [37] Rosenberg A. E., Effect of Glottal Pulse Shape on the Quality of Natural Vowels, *The Journal of the Acoustical Society of America*, 1971, **49**(2B), 583–590.
- [38] Juang B. H., Rabiner L. R., Automatic speech recognition—A Brief History of the Technology Development, Georgia Institute of Technology. Atlanta Rutgers University and the University of California. Santa Barbara, 2005, **1**, 67.
- [39] Fifty Years of Signal Processing. <https://signalprocessingsociety.org/uploads/history/history.pdf> (accessed date: Jun. 25, 2021).
- [40] Shannon C. E., A Mathematical Theory of Communication, *The Bell system technical journal*, 1948, **27**(3), 379-423.



- [41] Künzel H., Automatic Speaker Recognition of Identical Twins, *International Journal of Speech, Language & the Law*, 2011, **17**(2), 251–277.
- [42] Furui S., Recent Advances in Speaker Recognition, *Pattern recognition letters*, 1997, **18**(9), 859-872.
- [43] Büyük O., Telephone-Based Text-Dependent Speaker Verification. PhD thesis, 2011.
- [44] Mysak E. D., Pitch and Duration Characteristics of Older Males, *Journal of Speech and Hearing Research*, 1959, **2**(1), 46–54.
- [45] Robson D., The age you feel means more than your actual birthdate. <https://www.bbc.com/future/article/20180712-the-age-you-feel-means-more-than-your-actual-birthdate> (Accessed date: Apr. 22, 2021).
- [46] Skoog Waller, S., Eriksson, M., Sörqvist, P., Can you hear my age? Influences of Speech Rate and Speech Spontaneity on Estimation of Speaker Age, *Frontiers in psychology*, 2015, **6**, 978.
- [47] Mysak E. D., Pitch and Duration Characteristics of Older Males. *Journal of Speech and Hearing Research* , 1959, **2**(1), 46-54.
- [48] Minematsu N., Sekiguchi M., Hirose K., Automatic estimation of one’s age with his/her speech based upon acoustic modeling techniques of speakers,” in *2002 IEEE International Conference on Acoustics, Speech, and Signal Processing*, Orlando, FL, USA, 13-17 May 2002.
- [49] Spiegl, W., Stemmer, G., Lasarczyk, E., Kolhatkar, V., Cassidy, A., Potard, B., ... Nöth, E. Analyzing Features for Automatic Age Estimation on Cross-Sectional Data, In: Tenth Annual Conference of the International Speech Communication Association. Brighton, UK, 6-10 Sep. 2009.
- [50] Li M., Han K. J., Narayanan S., (Automatic Speaker Age and Gender Recognition Using Acoustic and Prosodic Level Information Fusion. *Computer Speech & Language*, 2013, **27**(1), 151-167
- [51] Ajmera J., Burkhardt F., Age and Gender Classification using Modulation Cepstrum, In: *Odyssey*, 2008, 25.
- [52] Muller C., Burkhardt F., Combining Short-Term Cepstral and Long-Term Pitch Features for Automatic Recognition of Speaker Age, *Eighth Annual Conference of the International Speech Communication Association*. Antwerp, Belgium, 27-31 Aug. 2007.
- [53] Büyük O., Arslan L. M., Age identification from voice using feed-forward deep neural networks, in *2018 26th Signal Processing and Communications Applications Conference (SIU). IEEE*, Çeşme, Izmir, Turkey, 2-5 May 2018.
- [54] Sadjadi S. O., Ganapathy S., Pelecanos J. W., Speaker age estimation on conversational telephone speech using senone posterior based i-vectors, in

2016 *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Shanghai, China, 20-25 Mar. 2016.

- [55] Novotný O., Plchot O., Matějka P., Mošner L., Glembek O., On the use of X-vectors for Robust Speaker Recognition, in *Odyssey 2018 The Speaker and Language Recognition Workshop*, Les Sables d'Olonne, France, 26-29 June 2018.
- [56] Snyder D., Garcia-Romero D., Povey D., Khudanpur S., Deep Neural Network Embeddings for Text-Independent Speaker Verification, in *Interspeech 2017*, Stockholm, Sweden, 20-24 Aug. 2017.
- [57] Snyder D., Garcia-Romero D., Sell G., Povey D., Khudanpur S., X-Vectors: Robust DNN Embeddings for Speaker Recognition, in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Calgary, Alberta, Canada, 15–20 April 2018.
- [58] Snyder D., NIST SRE 2016 Xvector Recipe. 2017.
- [59] Volín J., Tykalová T., Bořil T., Stability of Prosodic Characteristics Across Age and Gender Groups, in *Interspeech 2017*, Stockholm, Sweden, 20-24 Aug. 2017.
- [60] Bahari M. H., McLaren M., Van hamme H., Van Leeuwen D. A., Speaker Age Estimation Using i-Vectors, *Engineering Applications of Artificial Intelligence*, 2014, **34**, 99–108.
- [61] Huckvale M., Webb A., A Comparison of Human and Machine Estimation of Speaker Age, in *International Conference on Statistical Language and Speech Processing*. Springer, Cham, Budapest, Hungary, 24-26 Nov, 2015.
- [62] Ghahremani, P., Nidadavolu, P. S., Chen, N., Villalba, J., Povey, D., Khudanpur, S., Dehak, N., End-to-end Deep Neural Network Age Estimation, in *Interspeech 2018*, Hyderabad , India, 2-6 Sep. 2018,
- [63] Mahmoodi D., Marvi H., Taghizadeh M., Soleimani A., Razzazi F., Mahmoodi M., Age Estimation Based on Speech Features and Support Vector Machine,” in *2011 3rd Computer Science and Electronic Engineering Conference (CEEC)*, Colchester, UK, 13-14 July 2011.
- [64] Fedorova A., Glembek O., Kinnunen T., Matějka P., Exploring ANN Back-Ends for i-Vector Based Speaker Age Estimation., *Sixteenth Annual Conference of the International Speech Communication Association*. Dresden, Germany, 6-10 Sep. 2015.
- [65] Strand O. M., Egeberg A., Cepstral Mean and Variance Normalization in the Model Domain, *COST278 and ISCA Tutorial and Research Workshop (ITRW) on Robustness Issues in Conversational Interaction*, University of East Anglia, Norwich, UK, 30-31 Aug. 2004.

- [66] Bottou L., Stochastic Gradient Learning in Neural Networks, *Proceedings of Neuro-Nimes*, 1991, **91**(8), 12.
- [67] Zazo R., Nidadavolu P. S., Chen N., Gonzalez-Rodriguez J., Dehak N., Age Estimation in Short Speech Utterances Based on LSTM Recurrent Neural Networks, *IEEE Access*, 2018, 6, 22524–22530.
- [68] Mallouh A. A., *A Framework for Enhancing Speaker Age and Gender Classification by Using a New Feature Set and Deep Neural Network Architectures*, University of Bridgeport, 2017.
- [69] Markitantov M., Verkholyak O., Automatic Recognition of Speaker Age and Gender Based on Deep Neural Networks, in *International Conference on Speech and Computer*. Springer, Cham, Istanbul, Turkey, 20-25 Aug. 2019.
- [70] Chung J. S., Nagrani A., Zisserman A., VoxCeleb2: Deep Speaker Recognition, in *Interspeech 2018*, Hyderabad, India, 2-6 Sep. 2018.
- [71] Kwasny D., Hemmerling D., Joint gender and age estimation based on speech signals using x-vectors and transfer learning, *arXiv preprint arXiv:2012.01551*, 2020.
- [72] Kitagishi Y., Kamiyama H., Ando A., Tawara N., Mori T., Kobashikawa S., Speaker Age Estimation Using Age-Dependent Insensitive Loss, in *2020 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, Auckland, New Zealand, 7-10 Dec. 2020.
- [73] Kalluri S. B., Vijayasenan D., Ganapathy S., Automatic speaker profiling from short duration speech data, *Speech Communication*, 2020, **121**, 16–28.
- [74] Vergin R. O'Shaughnessy D., Pre-emphasis and Speech Recognition, In *Proceedings 1995 Canadian Conference on Electrical and Computer Engineering*, IEEE, 1995, **2**, 1062-1065.
- [75] Oppenheim A. V., *Discrete-Time Signal Processing*. Pearson Education India, 1999.
- [76] Osman M. M., Büyük O., Parabolic Filter Mel Frequency Cepstral Coefficient and Fusion of Features for Speaker Age Classification, *Sigma: Journal of Engineering & Natural Sciences / Mühendislik ve Fen Bilimleri Dergisi*, 2020, **38**( 4), 2177–2191.
- [77] Hanilci C., Features and Classifiers for Replay Spoofing Attack Detection, *2017 10Th international conference on electrical and electronics engineering (ELECO)*, IEEE, Bursa, Turkey, 30 Nov. - 02 Dec. 2017.
- [78] On C. K., Pandiyan P. M., Yaacob S., Saudi A., Mel-frequency cepstral coefficient analysis in speech recognition, In *2006 International Conference on Computing & Informatics*. IEEE, Kuala Lumpur, Malaysia, 6-8 June 2006.

- [79] Lu L., Zhang H. J., Jiang H., Content Analysis for Audio Classification and Segmentation, *IEEE Transactions on Speech and Audio Processing*, 2002, **10**(7), 504–516.
- [80] Kua J. M. K., Thiruvaran T., Nosratighods M., Ambikairajah E., Epps, J., Investigation of Spectral Centroid Magnitude and Frequency for Speaker Recognition, in *Odyssey 2010 The Speaker and Language Recognition Workshop*. Brno, Czech Republic, 28 June - 01 Jul. 2010.
- [81] Zeng Y., Wu Z., Falk T., Chan W., Robust GMM Based Gender Classification using Pitch and RASTA-PLP Parameters of Speech, in *2006 International Conference on Machine Learning and Cybernetics*, Dalian, China, 13-16 Aug. 2006.
- [82] Schluter R., Ney H., Using Phase Spectrum Information for Improved Speech Recognition Performance, in *2001 IEEE International Conference on Acoustics, Speech, and Signal Processing Proceedings*, Salt Lake City, UT, USA, 7-11 May 2001.
- [83] Hegde R. M., Murthy H. A., Gadde V. R. R., Significance of the Modified Group Delay Feature in Speech Recognition, *IEEE Transactions on Audio, Speech, and Language Processing*, 2007, **15**(1), 190–202.
- [84] Mladenović D., Feature Selection for Dimensionality Reduction, in *International Statistical and Optimization Perspectives Workshop Subspace, Latent Structure and Feature Selection*. Springer, Berlin, Heidelberg, Germany, 23-25 Feb. 2005.
- [85] Alpaydm E., *Introduction to Machine Learning*, 2<sup>nd</sup> ed. Cambridge, Mass.: MIT Press, 2010.
- [86] Wei H. L., Billings S. A., Feature Subset Selection and Ranking for Data Dimensionality Reduction, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2007, **29**(1), 162-166.
- [87] Reynolds D. A., Quatieri T. F., Dunn R. B., Speaker Verification Using Adapted Gaussian Mixture Models, *Digital Signal Processing*, 2000, **10**(1-3), 19-41.
- [88] Wang L., Khan L., Thuraisingham B., An Effective Evidence Theory Based K-Nearest Neighbour (KNN) Classification, in *2008 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology*, Sydney, Australia, 9-12 Dec. 2008.
- [89] López-Chau A., Cervantes J., López-García L., Lamont F. G., Fisher's Decision Tree, *Expert Systems with Applications*, 2013, **40**(16), 6283-6291.
- [90] Abdi H., Williams L. J., Principal Component Analysis, *WIREs Computational Statistics*, 2010, **2**(4), 433–459.

- [91] Slater M., Lagrange Multipliers Revisited, in *Traces and Emergence of Nonlinear Programming*, G. Giorgi and T. H. Kjeldsen, Eds. Basel: Springer, 2014, 293–306.
- [92] Pruitt W. E., Eigenvalues of Non-Negative Matrices, *The Annals of Mathematical Statistics*, 1964, **35**(4), 1797–1800.
- [93] Vizlay P., Juhár J., Pleva M., Modified Estimation of Between-Class Covariance Matrix in Linear Discriminant Analysis of Speech, in *2013 20th International Conference on Systems, Signals and Image Processing (IWSSIP)*. IEEE, Bucharest, Romania, 7-9 July 2013.
- [94] Ghannay S., Estève Y., Camelin N., Deleglise P., Evaluation of acoustic word embeddings, in *Proceedings of the 1st Workshop on Evaluating Vector-Space Representations for NLP*, Berlin, Germany, Aug. 2016.
- [95] Mak M. W., Lecture Notes on Factor Analysis and i-Vectors, *Technical Report and Lecture Note Series, Department of Electronic and Information Engineering*, The Hong Kong Polytechnic University, Feb. 2016.
- [96] Gupta R., Chen Y., *Theory and Use of the EM Algorithm*. Now Publishers Inc, 2011.
- [97] Staff T. A., “How tech giants are investing in artificial intelligence,” Tech Advisor. <https://www.techadvisor.com/feature/small-business/tech-giants-investing-in-artificial-intelligence-3788534/> (Accessed date Aug. 09, 2021).
- [98] Peddinti V., Povey D., Khudanpur S., A Time Delay Neural Network Architecture for Efficient Modeling of Long Temporal Contexts, *Sixteenth annual conference of the international speech communication association*. Dresden, Germany 6 - 10 Sep. 2015.
- [99] Waibel A., Hanazawa T., Hinton G., Shikano K., Lang K. J., “Phoneme recognition using time-delay neural networks,” *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 1989, **37**(3), 328–339.
- [100] Snyder D., Chen G., Povey D., MUSAN: A Music, Speech, and Noise Corpus, *arXiv preprint arXiv:1510.08484*, 2015.
- [101] Ko T., Peddinti V., Povey D., Seltzer M. L., Khudanpur S., A Study On Data Augmentation Of Reverberant Speech For Robust Speech Recognition, in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, New Orleans, LA, USA, 5-9 Mar. 2017.
- [102] Devroye L., Györfi L., Lugosi G., *A Probabilistic Theory of Pattern Recognition*, Springer Science & Business Media, 1997.
- [103] Duda R. O., Hart P. E., Stork D. G., *Pattern Classification*, John Wiley & Sons, 2012.

- [104] Do C. B., Batzoglou S., What is the Expectation Maximization Algorithm?, *Nat Biotechnol*, 2008, **26**(8), 897–899.
- [105] Cui R., Bucur I. G., Groot P., Heskes T., A Novel Bayesian Approach for Latent Variable Modeling from Mixed Data with Missing Values, *Statistics and Computing*, 2019, **29**(5), 977–993.
- [106] Verma N. K., Dwivedi S., Sevakula R. K., Expectation Maximization Algorithm Made Fast for Large Scale Data,” in *2015 IEEE Workshop on Computational Intelligence: Theories, Applications and Future Directions (WCI)*, Kanpur, India, 14-17 Dec. 2015.
- [107] Ioffe S., Probabilistic Linear Discriminant Analysis, in *European Conference on Computer Vision*. Springer, Berlin, Heidelberg, Germany, 7-13 May 2006.
- [108] Gamage K. W., Sethu V., Le P. N., Ambikairajah E., An i-Vector GPLDA System for Speech Based Emotion Recognition, in *2015 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA)*, The Hong Kong Polytechnic University, Hong Kong, 16-19 Dec. 2015.
- [109] Prince S. J. D., Elder J. H., Probabilistic Linear Discriminant Analysis for Inferences About Identity, in *2007 IEEE 11th International Conference on Computer Vision*, Rio de Janeiro, Brazil, 14-21 Oct. 2007.
- [110] Dehak N., Kenny P. J., Dehak R., Dumouchel P., Ouellet P., Front-End Factor Analysis for Speaker Verification, *IEEE Transactions on Audio, Speech, and Language Processing*, 2011, **19**(4), 788–798.
- [111] Cumani S., Laface P., e-Vectors: JFA and i-Vectors Revisited,” in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, New Orleans, LA, USA, 5-9 Mar. 2017.
- [112] Dehak N., Discriminative and Generative Approaches for Long-and Short-Term Speaker Characteristics Modeling: Application to Speaker Verification, Ph.D. Thesis. École de Technologie Supérieure, 2009.
- [113] Porcelli M., Toint P. L., BFO, A Trainable Derivative-free Brute Force Optimizer for Nonlinear Bound-constrained Optimization and Equilibrium Computations with Continuous and Discrete Variables, *ACM Trans. Math. Softw.*, 2017, **44**(1), 1–25.
- [114] Haykin S. S., *Neural networks and learning machines*, 3<sup>rd</sup> ed. New York: Pearson, 2009.
- [115] Povey D., Ghoshal A., Boulianne G., Burget L., Glembek O., Goel N., ... Vesely K., The Kaldi Speech Recognition Toolkit, *presented at the IEEE 2011 Workshop on Automatic Speech Recognition and Understanding*, Waikoloa, Hawaii, USA, 11-15 Dec. 2011.

- [116] Long short-term memory, [https://en.wikipedia.org/w/index.php?title=Long\\_short-term\\_memory&oldid=1022481054](https://en.wikipedia.org/w/index.php?title=Long_short-term_memory&oldid=1022481054) (Accessed date: May 12, 2021).
- [117] Bishop C. M., *Neural Networks for Pattern Recognition*, Clarendon Press, 1995.
- [118] Statistics Toolbox User's Guide for MATLAB, 1993-2004 by The MathWorks, Inc., 2004.
- [119] Mallouh A. A., Qawaqneh Z., Barkana B. D., New Transformed Features Generated by Deep Bottleneck Extractor and a GMM-UBM Classifier for Speaker Age and Gender Classification, *Neural Computing and Applications*, 2018, **30**(8), 2581-2593.
- [120] Yücesoy E., Speaker age and gender classification using GMM supervector and NAP channel compensation method, *Journal of Ambient Intelligence and Humanized Computing*, 2020, **11**(5),1-10.
- [121] Saini J., Mehra R., Power Spectral Density Analysis of Speech Signal using Window Techniques, *International Journal of Computer Applications* , 2015, **131**(14), 33-36.
- [122] Özaydın S., Examination of Energy Based Voice Activity Detection Algorithms for Noisy Speech Signals, *European Journal of Science and Technology*, 2019, 157–163.
- [123] Drugman T., Stylianou Y., Kida Y., Akamine M., Voice Activity Detection: Merging Source and Filter-based Information, *IEEE Signal Processing Letters*, 2015, **23**(2), 252-256.

## PERSONAL PUBLICATIONS AND ACHIEVEMENTS

- [1] **Osman M. M.,** Büyük O., Parabolic Filter Mel Frequency Cepstral Coefficient and Fusion of Features for Speaker Age Classification, *Sigma: Journal of Engineering & Natural Sciences / Mühendislik ve Fen Bilimleri Dergisi*, Oct. 2020, 38(4), 2177–2191.
- [2] **Osman M. M.,** Büyük O., Hanilçı C., A performance Evaluation of features for speaker age classification, *ISPEC 7<sup>th</sup> International Convergence on Engineering and Natural Sciences*, Izmir, Turkey, 8-10 May 2020.





## **RESUME**

Mohammed Muntaz OSMAN has graduated from Jimma University, Department of Electrical Engineering in July, 2008 with great distinction ranking first in the department and second in Technology Faculty. As part of Ethiopian ministry of science and higher education (MSHE) the then ministry of education plan, he has been selected to serve as a graduate assistant at Jimma University for two years. In July, 2012 he obtained his Master of Science (M.Sc.) degree in Communication and Information Systems Engineering with great distinction from Huazhong University of Science and Technology (HUST) in the city of Wuhan, China. He has offered various undergraduate and graduate courses since he returned to Jimma University and reinstated in his capacity as a lecturer. The researcher has also advised several projects and enjoyed the success of Jimma University during his five years of tenure. After three years of successful service as an academician, coordinator in communication engineering graduate and research programs he started pursuing his doctorate study at Kocaeli University, Department of Electronics and Communication Engineering, in Turkey. This researcher has published an article entitled “Parabolic Filter Mel Frequency Cepstral Coefficient and Fusion of Features for Speaker Age Classification“, in an indexed journal and another one entitled “A Performance Evaluation of Features for Speaker Age Classification”, in an international conference. He has also submitted another research article entitled “Effect of Number and Position of Frames in Speaker Age Estimation“, to an indexed and peer reviewed journal. It has recently been accepted for publication. Teaching, research and knowledge sharing has always been the central part of this researcher’s life. This researcher had once served as a physics teacher to grade 8 students while he was studying grade 9 in high school for himself.