Proceedings of the 1st Joint SLTU and CCURL Workshop (SLTU-CCURL 2020), pages 211–217
Language Resources and Evaluation Conference (LREC 2020), Marseille, 11–16 May 2020

Gender Detection from Human Voice Using Tensor Analysis

Prasanta Roy, Parabattina Bhagath and Pradip K. Das
Computer Science and Engineering Department, Indian Institute of Technology Guwahati
Guwahati, Assam, India
roy174101001@iitg.ac.in, bhagath.2014@iitg.ac.in, pkdas@iitg.ac.in

Abstract

Speech-based communication is one of the most preferred modes of communication for humans. The human voice contains several important information and clues that help in interpreting the voice message. The gender of the speaker can be accurately guessed by a person based on the received voice of a speaker. The knowledge of the speaker's gender can be a great aid to design accurate speech recognition systems. GMM based classifier is a popular choice used for gender detection. In this paper, we propose a Tensor-based approach for detecting the gender of a speaker and discuss its implementation details for low resourceful languages. Experiments were conducted using the TIMIT and SHRUTI dataset. An average gender detection accuracy of 91% is recorded. Analysis of the results with the proposed method is presented in this paper.

Keywords: speaker gender detection, tensor decomposition, method of moments, tensor power method, MFCCs

1. Introduction

Gender detection is one of the important problems in speaker and speech recognition domains. It has got significance because of the gain in popularity of voice-based systems like Alexa, Google Assistant, Cortana, Siri, etc. One of the applications of this is helping companies to provide better solutions. In speech recognition, it helps in improving the accuracy of recognition. It also has importance in sub-problems like age detection, emotion detection, speaker identification, etc. Research on the gender detection problem started in the early '90s. The problem was studied by using features like Linear Predictive Cepstral Coefficients (LPCCs), energy, Mel Frequency Cepstral Coefficients (MFCCs), etc. Konig and Morgan (Konig and Morgan, 1992) used LPCCs in their work to address this problem. In the system that was proposed, a multi-layer perceptron was employed for the classification of gender. As a result, this system achieved an accuracy of 84% on DARPA resource management database.

Neti (Neti and Roukos, 1997) proposed a GMM (Gaussian Mixture Model) based gender classification approach for an Air Travel Information System (ATIS) corpus. It was reported that 95% accuracy was obtained. This was an improvement over a simple pattern matching approach. MFCCs have widely accepted features in speaker characterization. They play an important role in GMM based systems that deal with gender recognition task. Tzanetakis (Tzanetakis and Cook, 2002) proposed a system that uses the above- mentioned features. The system was developed with gender classification and sports announcement facilities. Along with the techniques that are discussed, there are papers available on the same problem. In these systems, the pitch was used as a crucial feature. Several studies agree that modeling techniques like Convolutional Neural Networks (CNNs) (Doukhan et al., 2018), Expectation-Maximization (EM) (Yücesoy and Nabiyev, 2013), Hidden Markov Models (HMMs) (Parris and Carey, 1996), Support Vector Machine (SVM) classifiers (Jo et al., 2008) are successful in this area of research.

GMM-based classifiers and Expectation-Maximization (EM) have been used predominantly for modeling and parameter estimation, respectively. Most of the methods for estimating parameters of GMM are based on Maximum Likelihood Estimation (MLE), which has a drawback of getting stuck in a local optimum. So it needs to restart indefinitely to search for global optimum, and sometimes it may not find global optimum at all. As a result, the whole process of parameter estimation becomes very time-consuming.

In this paper, we have proposed an eigenvector-based approach to detect the gender from human voice using tensor analysis. We have used MFCCs as feature vector to form the feature vector space. Method of moments is used to build the tensor structure from the feature vector space for each gender. The tensor power method is applied to compute the eigenvectors from that tensor structure (Anandkumar et al., 2014). The proposed approach does not require multiple restarts but still provides 91% accuracy using Euclidean distance for evaluations.

2. Basic Understanding of Tensors

In this section, we will go through the basics of Tensors and related multi-linear algebra that are essential concepts to understand the tensor power method (Anandkumar et al., 2017) and its usefulness in parameter estimation of latent variable models. A comprehensive study about tensor is available in the work of Kolda (Kolda and Bader, 2009) and Sidiropoulos (Sidiropoulos et al., 2017), whereas a multilinear map and its notations can be found in the work of Lim (Lek-Heng Lim, 2005).

2.1. Tensor Preliminaries

Tensor is a multiway collection of numbers or an extension of a matrix in higher order. Vectors and Matrices are first-order and second-order tensors, respectively. In general, a p^{th} order tensor is an object that can be interpreted as a p-dimensional array of numbers. *Tensor order* is the number of dimensions of the tensor. Though the tensor can be of any order, we will describe tensor as a 3^{rd} order tensor structure in our experiments. For discussion, an N-way ten-

sor is the same as N-order tensor or vice versa. In terms of notation, a scalar is denoted by lower case letters $a \in \mathbb{R}$, vectors by bold lower case letter $\boldsymbol{a} \in \mathbb{R}^{\mathbf{I_1}}$, matrices by upper case bold letter $A \in \mathbb{R}^{\mathbf{I_1} \times \mathbf{I_2}}$ and for higher order tensor calligraphic letters are used $\mathcal{A} \in \mathbb{R}^{\mathbf{I_1} \times \mathbf{I_2} \times \ldots \times \mathbf{I_N}}$.

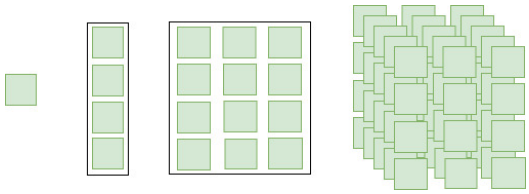

Figure 1: Zeroth Order Tensor ($a \in \mathbb{R}$, First Order Tensor ($\boldsymbol{a} \in \mathbb{R}^4$), Second Order Tensor ($A \in \mathbb{R}^{4\times3}$), Third Order Tensor ($\mathcal{A} \in \mathbb{R}^{4\times3\times5}$).

2.1.1. Outer Product and Inner Product

Vector outer product is the element-wise product of two vectors. The outer product of two vectors produces a Matrix, which is a second-order tensor. In this discussion, the outer product will be denoted by $\circledcirc$ symbol. For instance, if $\boldsymbol{a}$ and $\boldsymbol{b}$ are two n-sized vectors then their outer product will produce a matrix A as follows:

$$A = \boldsymbol{a} \circledcirc \boldsymbol{b} = \boldsymbol{a}\boldsymbol{b}^T \tag{1}$$

Similarly, the outer product of three vectors will generate 3^{rd} order tensor, which will be relevant to our topic of discussion. In general, the outer product of n vectors creates n-order tensor.

$$\mathcal{A} = \boldsymbol{a}^{(1)} \circledcirc \boldsymbol{a}^{(2)} \circledcirc \boldsymbol{a}^{(3)} ..... \circledcirc \boldsymbol{a}^{(n)} \tag{2}$$

In contrast to this, the inner product of two m-sized vectors will generate a scalar.

$$a = \boldsymbol{a}^T \boldsymbol{b} = \sum_{i=1}^{m} \boldsymbol{a}_i \boldsymbol{b}_i \tag{3}$$

2.1.2. Tensor Rank

Tensor rank is one of the important properties of a tensor. Before going to tensor rank, we will discuss about Rank-1 tensor. If an N-order tensor is strictly decomposed as an outer product of N vectors, then the N-order tensor is a Rank-1 tensor. So a Rank-1 matrix (2-way tensor) can be written as $A = \boldsymbol{a} \circledcirc \boldsymbol{b}$. Similarly a Rank-1- third-order tensor can be represented as $\mathcal{A} = \boldsymbol{a} \circledcirc \boldsymbol{b} \circledcirc \boldsymbol{c}$.
Minimum number of rank-1 N order tensors required that can sum up as N order tensor is called the rank of the N-order tensor. A rank-R third-order tensor can be represented as $\mathcal{A} = \sum_{i=1}^{R} \lambda_i \boldsymbol{a}_i \circledcirc \boldsymbol{b}_i \circledcirc \boldsymbol{c}_i$. Here the λ is used to represent the weighting factor during normalization of matrices, which are the other factors of the resultant tensor.

2.2. Tensor Decomposition

In Mathematics, it is fundamental to decompose an object into some simpler and easy-to-handle objects. Matrix decomposition techniques are significant in the field of Mathematics in their application to solve linear equation systems and the implementation of numerical algorithms efficiently.

In the following part, we have discussed the non-uniqueness of general matrix decomposition and the uniqueness of tensor decomposition with much-relaxed conditions.

2.2.1. Matrix Decomposition and Rotational Problem

In our discussion on matrix decomposition, we focus on matrix rank decomposition, which is an information extraction technique. It can be expressed by the following equation:

$$A = BC^T \tag{4}$$

where $A \in \mathbb{R}^{n\times m}, B \in \mathbb{R}^{n\times r}, C \in \mathbb{R}^{m\times r}$ and r is rank of the decomposition.
Similar work was carried out by Charles Spearman, a British Psychologist in 1904, which is popularly known as Spearman's Hypothesis.
However Equation 4 is not unique. By using another invertible matrix R, we can create another decomposition. Absorbing R on the left with B and R^{-1} on the right of C we can generate matrix $\dot{B}$ and $\dot{C}$ respectively which can be used to reconstruct A.

$$A = BC^T = BRR^{-1}C^T = (BR)(R^{-1}C^T) = \dot{B}\dot{C} \tag{5}$$

We can see that matrix rank-decomposition is non-unique generally. Though some decomposition techniques provide unique decomposition over some conditions such as orthogonality for Singular Value Decomposition (SVD), tensor decomposition is unique under much milder conditions.

2.2.2. Tensor Uniqueness and Rigidness

Tensor decomposition is unique only if there is one type of rank-1 tensor that sums up to our main tensor with a certain scaling factor. It means we cannot construct a different arrangement of rank-1 tensors that can sum up to our desired main tensor. The uniqueness of tensor decomposition is under much milder conditions than matrix decomposition. Let's consider a slice of a tensor $\mathcal{A}$ which can be represented as follows:

$$A_k = \sum_{i=1}^{R} (\boldsymbol{a}_i \circledcirc \boldsymbol{b}_i) \boldsymbol{c}_{ki} \tag{6}$$

Here k represents the k^{th} slice which is also a low-rank matrix. Therefore a tensor is not just a low-rank collection of these slices, there is an interrelation among them. If we observe, each slice is a differently scaled representation of the same matrix. This constraint helps us to address the rotational problem of a matrix that is faced during matrix decomposition.
To determine the factors that capture the underlying structure of a tensor, we subtract the scaled matrix formed by those factors. For matrices, there are multiple possibilities of finding those factors. But for tensors, these factors have to satisfy all the slices, thus making a strong interconnection between the slices, which further makes the tensor more rigid.

2.3. Tensor Decomposition Algorithms

Tensor Decomposition is one of the most studied topics of tensors. There are two different families of tensor decomposition techniques as follows:

1. Canonical Polyadic Decomposition (CPD)

2. Tucker Decomposition

CPD is mainly used for latent parameter estimation, and Tucker is used for compression, dimensionality reduction, estimation of subspace, etc.
In the following subsections, first, we have discussed the basic understanding of CPD and Tucker decomposition, followed by the tensor power method, which is a special kind of CPD decomposition. The tensor power method is used in our proposed approach.

2.3.1. Canonical Polyadic Decomposition

A rank decomposition is a way to express a tensor as a sum of rank-1 tensors of finite numbers. Rank decomposition has been discovered differently in different knowledge domains in many forms. Parallel Factors (PARAFAC) and Canonical Decomposition (CANDECOMP) is the most popular among them. The basic principle is the same for them. We will refer to this as CANDECOMP/PARAFAC or Canonical polyadic decomposition.
CPD for a 3-way Tensor($\mathcal{A}$) can be expressed as

$$\min_{\hat{\mathcal{A}}} \|\mathcal{A} - \hat{\mathcal{A}}\|$$

where

$$\hat{\mathcal{A}} = \sum_{i=1}^{R} \boldsymbol{a}_i \circledcirc \boldsymbol{b}_i \circledcirc \boldsymbol{c}_i \tag{7}$$

Different algorithms are available to compute the CPD of any given tensor. Jennrich's and Alternating Least Square Algorithm (ALS) are the most popular among them.
Let A, B and C be factor matrices that holds the combination of vectors $(\boldsymbol{a}_i, \boldsymbol{b}_i, \boldsymbol{c}_i)$ forming the rank-1 tensor $\mathcal{A}$ as columns.

$$A = [\boldsymbol{a}_1 \boldsymbol{a}_2 ... \boldsymbol{a}_R]$$

$$B = [\boldsymbol{b}_1 \boldsymbol{b}_2 ... \boldsymbol{b}_R]$$

$$C = [\boldsymbol{c}_1 \boldsymbol{c}_2 ... \boldsymbol{c}_R]$$

Jennrich's algorithm states that if A, B, and C are linearly independent, then the matrix have full rank. We can use this algorithm to compute the factor matrices as the tensor $\mathcal{A} = \sum_{i=1}^{R} \lambda_i \boldsymbol{a}_i \circledcirc \boldsymbol{b}_i \circledcirc \boldsymbol{c}_i$. It is unique up to a trivial permutation of rank and scaling factors. This algorithm works for some problem, but it does not consider all the tensor slices, and it also requires a good difference between two successive eigen values (eigen-gap), absence of which causes numerical instability.
ALS is state of the art for modern tensor decomposition techniques in the CPD family. The key idea is to fix all factor matrices for the tensor except one and then estimating the non-fixed matrix. This step is repeated for all the factor matrices until a specific stopping criterion is achieved. Though the ALS algorithm is straightforward, it takes several steps to converge, and sometimes it may also get stuck at a local optimum.

2.3.2. Tucker Decomposition

In this type of decomposition, a tensor is decomposed in a core tensor and factor matrices. Algorithms like Higher-Order Singular Value Decomposition (HOSVD), Higher-Order Orthogonal Iteration (HOOI) comes under this family of decomposition. However, in contrast to CPD, Tucker decomposition is not unique, and so it is not used for the estimation of latent variables.

2.3.3. Tensor Power Method

This method is a special type that comes under the CPD family. The tensors that can be decomposed by this algorithm should have the following structure:

$$\mathcal{A} = \sum_{i=1}^{R} \lambda_i \boldsymbol{a}_i \circledcirc \boldsymbol{a}_i \circledcirc \boldsymbol{a}_i \tag{8}$$

In this special case, the factor matrices have to be identical, and $\boldsymbol{a}_i$'s need to be orthogonal to construct vectors from rank-1 tensors. It is very similar to the matrix power method, but this algorithm tries to calculate top singular vectors in a tensor.
The main idea behind the matrix power method is to estimate the eigenvector $\boldsymbol{a}_{i,k+1}$ to $\boldsymbol{a}_i$ as well as the eigenvalue λ_i based on the following recurrence relation:

$$\boldsymbol{a}_{i,k+1} = \frac{A_i(I, \boldsymbol{a}_{i,k})}{\|A_i(I, \boldsymbol{a}_{i,k})\|_2} = \frac{A_i \boldsymbol{a}_{i,k}}{\|A_i \boldsymbol{a}_{i,k}\|_2} \tag{9}$$

where $\boldsymbol{a}_{i,0}$ will be chosen randomly, or it can be initialized with some correlation to the true eigenvector if possible.
This approximation follows the eigenvector/-value relationship $A\boldsymbol{a}_i = A(I, \boldsymbol{a}_i) = \lambda_i \boldsymbol{a}_i$. The top singular value can be computed from the computed eigenvector after convergence. As we have to calculate the first few dominant eigenvalues, this can be computed by the same process after deflating the matrix by the following formulae:

$$A_{i+1} = A_i - \lambda_i \boldsymbol{a}_i \circledcirc \boldsymbol{a}_i \tag{10}$$

To use this matrix power method in the Tensor approach, we have to incorporate the following changes in Equation (9).

$$\boldsymbol{a}_{i,k+1} = \frac{\mathcal{A}_i(I, \boldsymbol{a}_{i,k}, \boldsymbol{a}_{i,k})}{\|\mathcal{A}_i(I, \boldsymbol{a}_{i,k}, \boldsymbol{a}_{i,k})\|} \tag{11}$$

$$\mathcal{A}_{i+1} = \mathcal{A}_i - \lambda_i \boldsymbol{a}_i \circledcirc \boldsymbol{a}_i \circledcirc \boldsymbol{a}_i \tag{12}$$

This tensor Power method was used in the proposed method because of its efficiency in calculating the tensor. In the next section, the approach is explained in detail.

3. Proposed Approach

An uttered sound of a speaker is a collection of feature vectors. Each feature vector is a scaled sum of eigenvectors of that feature vector space. Some of these eigenvectors can be factors that represent age, gender, or other properties about the speakers while some form the content of the speech. If we collect feature vectors of male speaker utterances and construct a feature vector space from those, then that feature vector space gets dominated by the eigenvectors, which are the factors of masculinity. The same goes

for females. For any unknown utterances of the speaker, if we find the presence of these eigenvectors, we can infer the gender of the speaker.
The following part consists of feature vector space generation of each gender, computation of dominant eigenvectors using the tensor power method, and finding the presence of these eigenvectors in an unknown utterance.

3.1. Feature Vector Space Generation

We have used MFCCs as feature vectors to generate vector-space for each gender as MFCC is based on the principle of the human's auditory system. Twenty-six MFCCs are collected from each frame of an utterance. Thus each feature vector is of twenty six dimensions ($\boldsymbol{x} \in \mathbb{R}^{26}$). We have a collection of utterances for male and female speakers. We have computed feature vectors from each of the collections and obtained a set of feature vectors for each gender. This set of feature vectors works as a feature space that is used to compute dominant eigenvectors.

3.2. Tensor Formation

Before applying the tensor power method to compute the dominant eigenvectors, we have to form tensor from the feature vectors of each feature-space. A 3^{rd} order tensor is constructed from each set of feature vectors. Method of moments is used to construct the 3^{rd} order tensor. The first raw moment is the mean, which can be computed by the following:

$$\boldsymbol{m_1} = \mu = E[\boldsymbol{x}] = \frac{1}{N}\sum_{i=1}^{N} \boldsymbol{x}_i \tag{13}$$

where N is the number of feature vectors in each gender set.
Second ordinal moment can be computed by the following:

$$M_2 = E[\boldsymbol{x} \otimes \boldsymbol{x}] - \sigma^2 I \tag{14}$$

where σ^2 is the smallest eigenvalue of the covarience matrix ($\Sigma = E[\boldsymbol{x} \otimes \boldsymbol{x}] - \boldsymbol{m_1} \otimes \boldsymbol{m_1}$) and I is the Identity matrix ($I \in \mathbb{R}^{d \times d}$). Similarly the third ordinal moment can be computed as:

$$\begin{aligned}\mathcal{M}_3 = E[\boldsymbol{x} \otimes \boldsymbol{x} \otimes \boldsymbol{x}] - \sigma^2 \sum_{i=1}^{d} (\boldsymbol{m_1} \otimes \boldsymbol{e}_i \otimes \boldsymbol{e}_i \\ + \boldsymbol{e}_i \otimes \boldsymbol{m_1} \otimes \boldsymbol{e}_i + \boldsymbol{e}_i \otimes \boldsymbol{e}_i \otimes \boldsymbol{m_1})\end{aligned} \tag{15}$$

where $\boldsymbol{e}_i$ is the basis vector in i^{th} dimension.
From the work of Hsu and Kakade (Hsu and Kakade, 2013) these moments can be reduced to the following forms:

$$M_2 = \sum_{i=1}^{p} \boldsymbol{w}_i \boldsymbol{a}_i \otimes \boldsymbol{a}_i \tag{16}$$

$$\mathcal{M}_3 = \sum_{i=1}^{p} \boldsymbol{w}_i \boldsymbol{a}_i \otimes \boldsymbol{a}_i \otimes \boldsymbol{a}_i \tag{17}$$

Thus $\mathcal{M}_3$ is the scaled sum of p eigenvectors ($\boldsymbol{a}_i$). We need to find the k dominant eigenvectors that are responsible for the gender property of the speaker. M_2 could have been used to compute $\boldsymbol{a}_i$s, but due to matrix rotational problem, it can not be computed accurately. Whereas in tensor (3^{rd} order or higher), these can be computed more easily.
These Eigenvectors ($\boldsymbol{a}_i$) can be computed by the tensor power method only if they are orthogonal in nature. For that, we have to orthogonalize $\mathcal{M}_3$. This has been done using M_2. It is assumed that if a Matrix is found that can orthogonalize M_2 can help to orthogonalize $\mathcal{M}_3$. This orthogonalization of M_2 can be represented as:

$$M_2(W, W) = W^T M_2 W = I \tag{18}$$

where W is the orthogonalizing matrix, It is also known as the whitening matrix. W can be calculated with the help of eigenvalue decomposition of second-order moment M_2:

$$M_2 = U D U^T \tag{19}$$

Singular value decomposition has been used to find U, D from Equation (19). W is computed as follows:

$$W = U D^{\dagger \frac{1}{2}} \tag{20}$$

where $U \in \mathbb{R}^{d \times k}$ is a matrix of orthonormal eigenvectors, $D \in \mathbb{R}^{k \times k}$ is a diagonal matrix of the eigenvalues of M_2 and $A^\dagger$ is the Moore-Penrose pseudoinverse of matrix A. By using the following formulae W transforms $\mathcal{M}_3$ into whitened space.

$$\widehat{\mathcal{M}_3} = \mathcal{M}_3(W, W, W)^1 = \sum_{i=1}^{k} \lambda_i \boldsymbol{v}_i \otimes \boldsymbol{v}_i \otimes \boldsymbol{v}_i \tag{21}$$

where $\boldsymbol{v}_i$ and λ_i are converted eigenvectors and scaling factors respectively after orthogonalization of $\mathcal{M}_3$.

3.3. Eigenvectors Computation

Now on $\widehat{\mathcal{M}_3}$ we have applied tensor power method to identify dominant eigenvectors ($\boldsymbol{v}_i$). We shall use Equation (11) and Equation (12) to compute the $\boldsymbol{v}_i$s and deflate the tensor, respectively. This process will be repeated until k dominant eigenvectors are obtained. As $\boldsymbol{v}_i$s are computed from orthogonalized tensor ($\widehat{\mathcal{M}_3}$), so by applying the inversion of the orthogonalization process we transform $\boldsymbol{v}_i$s to $\boldsymbol{a}_i$s of M_3. We shall use the following formulae to do so:

$$A = (W^T)^\dagger V Diag(\lambda) \tag{22}$$

where A is the set of k number of $\boldsymbol{a}_i$s, V is the set of k number of $\boldsymbol{v}_i$ and λ_i are k eigenvalues computed from the tensor power method.

---

[1] A k^{th} order tensor is denoted by $\mathcal{A} = [\![\boldsymbol{a}_{j_1 \dots j_k}]\!] \in \mathbb{R}^{d_1 \times \dots \times d_k}$. Then covariant multi-linear matrix multiplication of $\mathcal{A}$ by $M_1 = [m^{(1)}_{j_1 i_1}] \in \mathbb{R}^{d_1 \times p_1}, \dots, M_k = [m^{(k)}_{j_1 i_1}] \in \mathbb{R}^{d_k \times p_k}$ can be defined as: $\mathcal{A}(M_1, \dots, M_k) = \left[\!\!\left[\sum_{j_1=1}^{d_1} \dots \sum_{j_k=1}^{d_k} a_{j_1 \dots j_k} m^{(1)}_{j_1 i_1} \dots m^{(k)}_{j_k i_k} \right]\!\!\right] \in \mathbb{R}^{p_1 \times \dots \times p_k}$

3.4. Model Creation and Evaluation

We have obtained k dominant eigenvectors from each of the feature vector set of male and female speakers. A_m and A_f are the eigenvectors set of male and female speaker, respectively.

For any unknown the feature vector in the feature space, we will calculate distance from the dominating eigenvector (minimum distance). The distance for i^{th} feature vector ($\boldsymbol{x}_i$) is calculated by using the following formula:

$$D_i = \min_k (\sum_{j=1}^{d} (a_{kj} - x_{ij})^2) \tag{23}$$

Total distance from A_f and A_m can be computed as follows:

$$\mathcal{D}_m = \sum_{i=1}^{N} D_i \tag{24}$$

$$\mathcal{D}_f = \sum_{i=1}^{N} D_i \tag{25}$$

where N is the total number of feature vectors (Number of frames) for a voice sample.

Features vectors collected from male voice will be containing vectors which are affected by male eigenvectors, whereas it will be less affected by the female eigenvectors. Thus $\mathcal{D}_m$ will be less than $\mathcal{D}_f$. For similar reasons, $\mathcal{D}_f$ will be less than $\mathcal{D}_m$ for the female voice.

4. Experimental Setup

Experiments were conducted on two different datasets (TIMIT (S Garofolo et al., 1992) and SHRUTI (Das et al., 2011)). The study can be divided into three different cases, as follows:

1. TIMIT DR1
2. TIMIT Mix
3. SHRUTI dataset

The first dataset is a subset of the TIMIT dataset, which consists of only the New England dialect. TIMIT Mix dataset is the subset that contains eight different dialect regions. The third dataset is a collection of spoken sentences belonging to the Bengali language. Bengali is the predominant language used in West Bengal, a state of the Indian subcontinent. In the present work, a subpart of this database was used. Table 4. gives the complete description of the dataset used in the study. The results obtained using the approach are discussed in the next section.

Dataset Type	Training Set		Testing Set	
	Male	Female	Male	Female
TIMIT (DR1)	246	146	34	25
TIMIT Mix	500	500	150	150
SHRUTI	650	650	150	150

Table 1: Description of datasets.

5. Results and Analysis

The results are presented for different cases, as follows:

1. Different sizes of feature vectors
2. Different number of eigenvectors
3. Comparison on multiple datasets
4. Evaluation of same trained models for different datasets
5. Performance evaluation on noisy data
6. Performance comparison with GMM-EM

At first, feature vectors were used with varying sizes of thirteen, twenty, and twenty-six, while each case considers four dominant eigenvectors. A significant amount of increment in gender detection is observed with the increase of feature vector size. It implies that the proposed approach can capture sufficient characteristics of gender properties successfully. The summary of the results is shown in Table 2.

Size of feature vectors	Dataset Type	Accuracy Type (%)		
		Male	Female	Average
13	**Training**	71.2	98.4	84.7
	Testing	70.4	97.1	83.5
20	**Training**	92.2	76.8	84.5
	Testing	95.2	72.8	84.0
26	**Training**	90.8	92.4	91.2
	Testing	93.36	89.82	91.59

Table 2: Performance with respect to different sizes of feature vector (d).

Next, the performance of the proposed approach with respect to different numbers of dominating eigenvectors was evaluated. In this experiment, the TIMIT Mix dataset was used. The results are shown in Table 3. This experiment also shows that there is an increment in average gender detection accuracy, which denotes that the eigenvectors computed by the proposed approach are relevant to gender detection.

Number of Eigenvectors	Dataset Type	Accuracy Type (%)		
		Male	Female	Average
1	**Training**	46.4	80.2	63.3
	Testing	42.03	76.10	59.06
2	**Training**	72.4	86.4	79.4
	Testing	75.66	84.84	84.75
3	**Training**	90.0	92.4	91.2
	Testing	92.92	91.15	92.03
4	**Training**	90.8	92.4	91.2
	Testing	93.36	89.82	91.59

Table 3: Performance with respect to the number of dominant eigenvectors (k).

We tested the performance of the proposed approach in different datasets: SHRUTI, TIMIT Mix, and TIMIT DR1. Figure 2 shows that the proposed method provides consistent performance across different datasets. To test whether

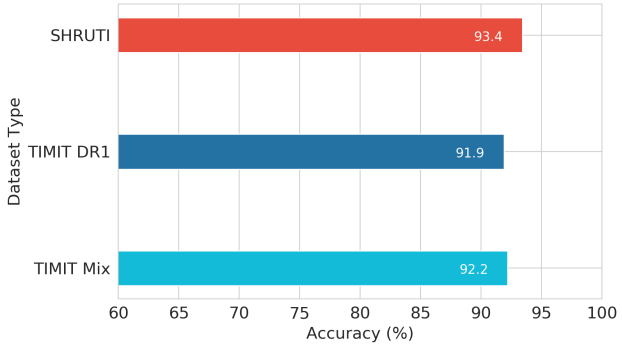

Figure 2: Performance of the proposed approach for different datasets.

the proposed approach is capturing the language-specific or voice-specific gender property, we computed eigenvectors using TIMIT Mix dataset and evaluated with other datasets. We have obtained a comparable accuracy in different datasets (Figure 3), which demonstrates that the proposed approach captures the voice-specific gender property.

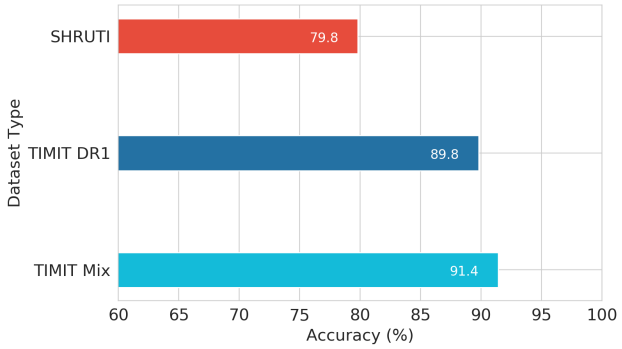

Figure 3: Performance of the proposed approach for different datasets trained using single dataset.

We evaluated its performance with respect to noisy utterances. Figure 4 shows the performance of the proposed approach with different Signal to Noise Ratio (SNR). The proposed method provides a consistent performance where the SNR is more than ten for input utterances.

Size of feature vector	Accuracy (%)	
	GMM - EM	Proposed approach
13	93.2	84.1
26	97.4	91.4

Table 4: Performance comparison of GMM and the proposed approach.

We also compared the performance of our approach with the modern, state-of-the-art GMM-EM on the TIMIT

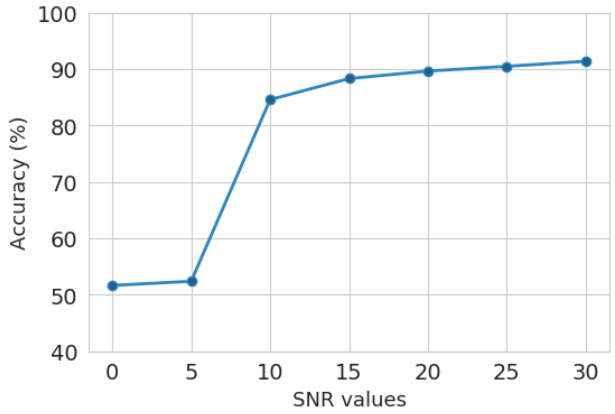

Figure 4: Performance of the proposed approach with respect to noisy data.

dataset. We conducted this experiment on the feature vector of size thirteen and twenty-six. We have presented our results in Table 4.

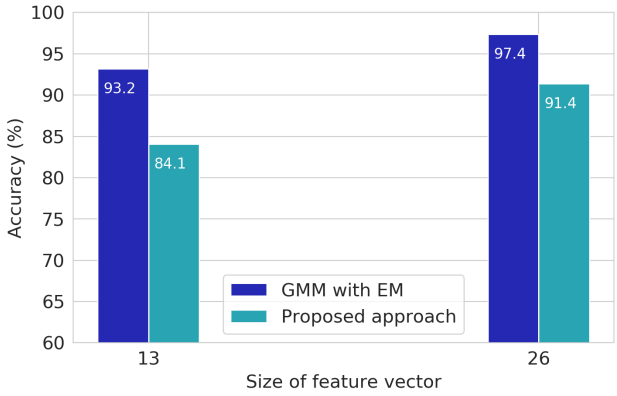

Figure 5: Performance comparison of GMM and the proposed tensor based approach.

Figure 5 provides a comparison between GMM-EM and the proposed method. Even though the detection efficiency of the proposed approach is comparatively less, but the proposed approach does not require multiple restarts like GMM-EM, and the improvement of results with the varying feature vectors is encouraging.

6. Conclusion

In this paper, a simple yet effective tensor-based approach was proposed for gender detection from the human voice. In the approach, we have computed dominant eigenvectors of the feature space of utterances using tensor analysis. It is demonstrated that the proposed method captures the relevant gender properties of the human voice and also provides consistent performance for high dimensional feature vectors. We have evaluated this approach on different datasets and proved that its performance is consistent with an accuracy of 91% in each case. We have also demonstrated its performance on noisy data and concluded that it provides reasonable accuracy for SNR higher than 10. The proposed approach provided comparable performance with respect to

GMM-EM, which ensures that with further improvement, and it can offer better performance without the drawbacks of GMM-EM. This work shows that the eigenvector-based approach using tensor analysis provides consistent performance irrespective of the dataset.

7. Bibliographical References

Anandkumar, A., Ge, R., Hsu, D., Kakade, S. M., and Telgarsky, M. (2014). Tensor decompositions for learning latent variable models. *J. Mach. Learn. Res.*, 15(1):2773–2832, January.

Anandkumar, A., Ge, R., and Janzamin, M. (2017). Analyzing tensor power method dynamics in overcomplete regime. *J. Mach. Learn. Res.*, 18(1):752–791, January.

Das, B., Mandal, S., and Mitra, P. (2011). Bengali speech corpus for continuous auutomatic speech recognition system. In *2011 International Conference on Speech Database and Assessments (Oriental COCOSDA)*, pages 51–55, Oct.

Doukhan, D., Carrive, J., Vallet, F., Larcher, A., and Meignier, S. (2018). An open-source speaker gender detection framework for monitoring gender equality. April.

Hsu, D. and Kakade, S. M. (2013). Learning mixtures of spherical gaussians: Moment methods and spectral decompositions. In *Proceedings of the 4th Conference on Innovations in Theoretical Computer Science*, ITCS '13, pages 11–20, New York, NY, USA. ACM.

Jo, Q., Park, Y., Lee, K., and Chang, J. (2008). A support vector machine-based voice activity detection employing effective feature vectors. *IEICE Transactions*, 91-B(6):2090–2093.

Kolda, T. G. and Bader, B. W. (2009). Tensor decompositions and applications. *SIAM Rev.*, 51(3):455–500, August.

Konig, Y. and Morgan, N. (1992). Gdnn: a gender-dependent neural network for continuous speech recognition. In *[Proceedings 1992] IJCNN International Joint Conference on Neural Networks*, volume 2, pages 332–337 vol.2, June.

Lek-Heng Lim. (2005). Singular values and eigenvalues of tensors: a variational approach. In *1st IEEE International Workshop on Computational Advances in Multi-Sensor Adaptive Processing, 2005.*, pages 129–132, Dec.

Neti, C. and Roukos, S. (1997). Phone-context specific gender-dependent acoustic-models for continuous speech recognition. In *1997 IEEE Workshop on Automatic Speech Recognition and Understanding Proceedings*, pages 192–198, Dec.

Parris, E. S. and Carey, M. J. (1996). Language independent gender identification. In *1996 IEEE International Conference on Acoustics, Speech, and Signal Processing Conference Proceedings*, volume 2, pages 685–688 vol. 2, May.

S Garofolo, J., Lamel, L., M Fisher, W., Fiscus, J., S. Pallett, D., L. Dahlgren, N., and Zue, V. (1992). Timit acoustic-phonetic continuous speech corpus. *Linguistic Data Consortium*, 11.

Sidiropoulos, N. D., De Lathauwer, L., Fu, X., Huang, K., Papalexakis, E. E., and Faloutsos, C. (2017). Tensor decomposition for signal processing and machine learning. *IEEE Transactions on Signal Processing*, 65(13):3551–3582, July.

Tzanetakis, G. and Cook, P. (2002). Musical genre classification of audio signals. *IEEE Transactions on Speech and Audio Processing*, 10:293–302, Jan.

Yücesoy, E. and Nabiyev, V. (2013). Gender identification of a speaker using mfcc and gmm. pages 626–629, Nov.