Dabbabi et al. EURASIP Journal on Audio, Speech, and Music Processing (2017) 2017:21
DOI 10.1186/s13636-017-0117-1

EURASIP Journal on Audio, Speech, and Music Processing

RESEARCH Open Access

Integration of evolutionary computation algorithms and new AUTO-TLBO technique in the speaker clustering stage for speaker diarization of broadcast news

Karim Dabbabi[1*], Salah Hajji[2] and Adnen Cherif[1]

Abstract

The task of speaker diarization is to answer the question "who spoke when?" In this paper, we present different clustering approaches which consist of Evolutionary Computation Algorithms (ECAs) such as Genetic Algorithm (GA), Particle Swarm Optimization (PSO) algorithm, and Differential Evolution (DE) algorithm as well as Teaching-Learning-Based Optimization (TLBO) technique as a new optimization technique at the aim to optimize the number of clusters in the speaker clustering stage which remains a challenging problem. Clustering validity indexes, such as Within-Class Distance (WCD) index, Davies and Bouldin (DB) index, and Contemporary Document (CD) index, is also used in order to make a correction for each possible grouping of speakers' segments. The proposed algorithms are evaluated on News Broadcast database (NDTV), and their performance comparisons are made between each another as well as with some well-known clustering algorithms. Results show the superiority of the new AUTO-TLBO technique in terms of comparative results obtained on NDTV, RT-04F, and ESTER datasets of News Broadcast.

Keywords: Speaker diarization, PSO algorithm, GA algorithm, DE algorithm, TLBO technique, EA algorithms, Clustering validity index, DER

1 Introduction

Nowadays, the fast progress in multimedia sources make the use of archived audio documents an increasing need for efficient and effective means of searching and indexing through voluminous databases. In order to facilitate the access to the recording in audio databases, searching and tagging based on who is speaking can be at the top of many basic components required for dealing with audio archives, such as recorded meetings or an audio portion of News Broadcast shows.

The old approaches in speaker recognition are developed for speakers' identification and verification in a speech sample pronounced by one person. However, the basic recognition approach has to be extended to include both speaker detection and tracking in multi-speaker audio. In this work, we highlight the speakers' indexation and research in audio broadcast news (NDTV) for speaker diarization task. Indeed, speaker diarization is one of the speaker-based processing techniques in which the feature representation of the acoustic signal aims to represent the speaker information and discriminate between different talkers. It has been introduced in the NIST project of Rich Transcription in "who spoke when" evaluations [1]. According to the first definition in 1999 NIST Speaker Recognition evaluation, the identification of audio regions based on a given speaker is a tracking speaker task [2]. Concerning the speaker detection task in audio data, it is performed by diarization and tracking procedures and it has an

* Correspondence: dabbabikarim@hotmail.com
[1]Research Unity of Analysis and Processing of Electrical and Energetic systems, Faculty of Sciences of Tunis, University of Tunis El Manar, 2092 Tunis, Tunisia
Full list of author information is available at the end of the article

objective to make speaker-based indexation according to the detected speaker and ensure good retrieve of speaker-based information in audio recording. For the speaker diarization task, it aims to structure audio documents into speaker turns and give their true identities so that we can make an automatic transcription.

In the speaker diarization task, there is any prior knowledge about the speakers and their number. It consists of two main phases: the first one is a segmentation phase in which the speech is segmented into many smaller segments at the detected change points in a recording. Ideally, each small segment contains speech from just one speaker. The second phase is a clustering phase, which makes the clustering of the neighboring segments uttered by the same speaker. Currently, a bottom-up approach known as Hierarchical Agglomerative Clustering (HAC) is the most popular method for clustering [3]. Speaker diarization has been applied in several speech areas [4]. The transcription of telephone and broadcast meetings, auxiliary video segmentation, and dominant speaker detection represent its main applications. The alleviation of the amount of speech document management tasks can be performed by such an effective tool like speaker clustering [5, 6]. This latter can group and attribute similar audio utterances to the same speaker in audio document by some distance measures and clustering schemes in an unsupervised condition [7]. In previous years, spectral clustering has been proved have better effect than hierarchical clustering in speaker clustering [8, 9]. This is due to greedy research for hierarchical clustering, which has high computation complexity and produces a suboptimal solution. In contrast, spectral clustering has relative lower computation complexity and can produce a global solution.

Searching for suitable model which can represent short segments and enable a similarity and difference measure between neighboring segments for clustering represent an open search topic. Previous approaches have been used to model each segment with a single GMM model or I-vectors extracted from a Universal Background Model (UBM) like it has been described in [10]. Indeed, a Gaussian Mixture Model (GMM) adapted from the UBM which has been used to form an I-vector represents the state-of-the-art systems to represent segments. Generally, good results have been reported using UBM/I-vector. In [11], deep neural network (DNN) has been trained to construct UBM and T-matrix in order to make the extracted I-vectors better models of the underlying speech. This method has shown capability to construct accurate models of speech, even for short segments. This system has also achieved a significant improvement on the NIST 2008 speaker recognition evaluation (SRE) telephone task data compared to state-of-the-art approaches. In this work, we have tried to optimize the clustering of the extracted I-vectors using evolutionary algorithms (EAs), and teaching-learning-based optimization (TLBO) technique.

Speaker clustering based on feature vector distance employs the distance of samples to measure the similarity of two speech segments. Thus, a two-step clustering based on Cross Likelihood Ratio (CLR) has been used by some researchers at the aim to measure the similarity between segments [10]. This approach has been shown its effectiveness to resolve the problem of a single Gaussian model describing the complex distribution of the features. Also, in [12], Rand index has shown good efficiency by reducing the overall clustering errors when it has been used to measure the similarity between utterances. During the agglomeration procedure, Bayesian Information Criteria (BIC) can only make each individual cluster as homogenous as possible, but it cannot guarantee that the homogeneity for all clusters can finally be summed to reach a maximum [12]. In this work, we have used EAs at the aim to optimize the generated clusters and the required number of clusters by estimating and minimizing the clustering validity indexes (criteria). These metrics reflect the clustering errors that arise when utterances from the same speaker are clustered in different clusters or when utterances from different speakers are clustered in the same cluster. We approximate the clustering validity index by a function of similarity measures between utterances and then use the EAs to determinate the cluster in which each utterance should be located, such that function is minimized.

For the clustering stage, there are many techniques used to regroup unlabeled dataset into groups of similar objects called clusters. Indeed, the integration of the evolutionary computation (EC) techniques by researchers in object clustering has an objective to develop clusters in complex dataset. In addition, the EAs are general stochastic search methods which have been at first applied in the biological world simulating natural selection and evolution. Also, they are not limited to keep only one solution for a problem, but they are extended to conserve a population of potential solutions for a problem. Therefore, the EA algorithms have many advantages compared to other traditional search and classification techniques, such as they need less domain-specific information and they can be used easily on a set of solutions (they so-called population). Also, the EA algorithms are so popular in many fields of applications especially in pattern recognition and they include many algorithms, such as genetic algorithm (GA), particle swarm optimization (PSO), evolution programming (EP),

evolution strategies (ES), and differential evolution (DE) algorithm. A common concept based on simulating the evolution of the individuals that form the population using a predefined set of operators is a shared concept by all these algorithms. Therefore, the selection and search operators are the two kinds of operators commonly used. For the mutation and recombination, they constitute the most widely used search operators. To determine the optimal number of clusters, the within-class distance (WCD), Davies and Bouldin (DB), and contemporary document (CD) clustering validity indexes have been used in this work at the aim to provide global minima/maxima at the exact number of classes in the dataset. Thus, the quantitative evolution with global clustering validity index permits a correction of each possible grouping. For the evolution process, it starts by the domination of the best solutions in the population and the elimination of the bad ones. After that, the evolution of solutions converges when the near optimal partitioning of the dataset is represented by the fittest solution with the respect of the employed clustering validity index. By this way, in only one run of the evolutionary optimization algorithm, the optimal number of classes along with the accuracy cluster center coordinates can be located. In fact, the performance of the evolutionary optimization algorithm relies sharply on the selection of the clustering validity index.

For the GA, it has been first applied in 1975 by Holland [13] and it is well known in many application fields as a new tool for complex systems optimization. Its main feature is represented by its capability to avoid local minima. Also, the GA is an unsupervised optimization method which can be used freely without any constraint to find the best solution. Therefore, it is the most popular EA and it is well known for resolving hard optimization problems. The GAs have shown their best performance in many application areas, such as pattern recognition, image processing, and machine learning [14]. Comparing GAs to EP and ES techniques, these latter techniques have performed better than GAs for real-valued function optimization. In the speaker diarization research area, there are some works recorded using GA, such as that one in [15] where the GA has been explored to design filter bank in feature extraction method destined for speaker diarization application. Also, in [16], the feature dimension reduction has been made through GAs in the objective to speed up speaker recognition task. In this work, we have used both GA binary and real-coded representations beside different variation of the major control parameters like selection, crossover, and different distance measures of the fitness function using WCD, DB, and CD clustering validity indexes. These indexes have been also explored by PSO and DE algorithms.

Concerning the PSO algorithm, it is a population-based stochastic optimization technique which has been developed in [17, 18]. This algorithm simulates the social behavior of bird stocking or fish schooling. Its first applications have been performed to optimize clustering results in mining tasks. Also, it has been applied for clustering task in wireless sensor networks in which it has been shown its robustness comparing to random search (RS) and simulated annealing (SA) [19]. In addition, PSO algorithm has been tested in document clustering and more compact clusters has been generated by hybridizing PSO algorithm with k-means comparing to the use of k-means algorithm alone [20]. Therefore, the combination of k-means algorithm with PSO algorithm for data clustering has demonstrated high accuracy and fast convergence to optimum solution [21]. In speaker diarization field, PSO algorithm has known many applications, such as it has been used with mutual information (MI) in multi-speaker environment [22]. In 2009, PSO algorithm has been also used with SGMM algorithm in text-independent speaker verification, and good performance has been registered using both algorithms compared to SGMM algorithm alone [23]. In 2011, PSO algorithm has been exploited to encode possible segmentations of an audio record by computing a measure as a fitness function of PSO algorithm between the obtained segments and the audio data using MI. This algorithm has shown good results in all test problems effectuated on two datasets which contain up to eight speakers [22]. Moreover, an optimization of artificial neural network (ANN) for speaker recognition task using PSO algorithm has been performed and shown an improvement in performance comparing to the use of ANN algorithm alone [24]. Like other algorithms, the global PSO algorithm has its drawbacks which are summarized in its tendency to trapper in local optimum under some initialization conditions [25]. More information about the PSO variants as well as about its applications can be found in [26, 27]. Concerning the DE algorithm, it needs little or no parameter to tune for numerical optimization as well as it has shown good performance [5]. Also, this algorithm is characterized by its small parameters to be determinate, high convergence speed, and hardness to fall in local optimum [28]. The previous applications of this approach in real-world and artificial problems have shown that it is superior to GA and PSO algorithms in single objective, noise free, and numerical optimization. One among few works which have been carried out using DE algorithm in speaker recognition applications has been oriented to optimize GMM parameters [29]. In this work, GA,

PSO, and DE algorithms have been applied and compared to new TLBO optimization technique, which has been used for automatic clustering of large unlabeled dataset. Indeed, the TLBO technique does not need any prior information about the data to be classified, and it can find the optimal number of data partitions in some iterations. Therefore, this algorithm can be defined as a population-based iterative learning and it possesses more common characteristics than other EC algorithms. Indeed, this technique has shown more improvement in convergence time for solving an optimization problem in real-world real-time applications compared to GA, PSO, DE, and artificial bee colony (ABC) algorithms. In [30], an investigation has been performed about the effect of the introduction of the elitist concept in TLBO algorithm on the performance. In addition, another investigation about the common controlling parameters (population size and the number of generations) and their effects on the performance of the algorithm has been performed too. Moreover, the TLBO technique has been used in [31] in order to optimize four truss structures. In [32], the introduction of the concepts of number of teachers, adaptive teaching factor, tutorial training, and self-motivated learning has been proposed at the aim to improve the performance of the TLBO algorithm. In [33], the θ-multi-objective TLBO algorithm has been presented in the purpose of resolving the dynamic economic emission dispatch problem. Therefore, for the purpose of global optimization problems, a dynamic group strategy has been suggested in [34] in order to improve the performance of the TLBO algorithm too. In addition, the ability of the population has been explored in the original TLBO technique by introducing a ring neighborhood topology [35]. In [36], it has been considered that TLBO technique is one of the simplest and most efficient techniques, as it has been empirically shown to perform well on many optimization problems. From our knowledge, there is any work recorded in speaker diarization research area using TLBO algorithm. More details about the basis TLBO concept can be found in [37].

The remained sections of this paper are organized as follows: In Section 2, we explain the different components of our proposed model. Concerning the next section, we discuss the experimental results. In Section 4, we conclude our paper with fewer discussions.

2 Overview of the methodology

Our model consists of many phases, and a detailed description of each phase is given in the following sub-sections.

2.1 Feature extraction (MFCCs)

Only the first 19 Mel Frequency Cepstral Coefficient (MFCC) features have been used in the Speech Activity Detector (SAD) module, speaker segmentation module, and speaker clustering module. Beside these features, the short-time energy (STE) and the zero-crossing ratio (ZCR) plus the first- and second-order derivatives of MFCCs have been employed in the SAD module. Also, for the speaker segmentation, only 19 MFCCs and short-time energy (STE) have been used, whereas in the speaker clustering stage, the first- and second-order derivatives of MFCCs have been added. The frame sizes for the analysis windows were set to 30 ms with 20 ms frame overlap. For the sampling frequency, it was set to 16 KHz.

2.2 SAD

This subsystem was used for both silence and music removal modules. For the silence removal module, the silence was suppressed from the whole audio recording using energy-based bootstrapping algorithm followed by an iterative classification. After the removal silence, the identification of music and other audible no-speech sounds from the recording have been performed using music vs. speech bootstrap discriminator, which consists to train music model from frames, which are identified as music and have high confidence level. Thus, the music model is refined iteratively. For both silence and music removal modules, in order to avoid the sporadic no-speech to speech transitions, only the segments with more than 1 s duration has been considered as no-speech.

2.2.1 Silence removal

This phase has been performed by concatenating 19 MFCC features plus their first and second derivatives with STE. Each frame has been attributed to silence or speech classes according to a confidence value of energy. Thus, the frames with 20%, the lowest energies are called high-confidence silence frames, and the frames with 10%, the highest energies are called high-confidence speech frames. A Gaussian mixture of size 4 over the 60-dimensional feature space has been used to train bootstrap silence model. The same size has been also employed to train bootstrap speech model using speech frames, which have high confidence level of energy. An iterative classification is employed to perform the frame classification into speech or silence classes. The remained frames between these frames which have high confidence level of energy have been used to train silence and speech models at the next iteration. Increasing the number of iterations engenders an increase in the number of 60-dimensional Gaussians employed to model the

speech and silence GMMs till the maximum. The Gaussian Mixtures Model (GMM) with 32 components for the speech and 16 components for no-speech have been given the best results for silence and pauses removal. Also, the high-energy no-speech named the audible no-speech, such as music and jingles, have been classified as speech because the MFCCs and frames energy for music are more similar to speech more than silence.

2.2.2 Music removal

The frames which have high confidence level from the histogram of ZCR for music and from the histogram of energy for the speech are used to train both music and speech models in order to estimate their initial models. Thus, only 40% of the highest zero-crossing rate frames from the ZCR histograms are used as high-confidence music frames and train the music model. After that, a refinement of speech and music classes has been performed in order to discard only music segments in the iterative classification. Thus, this refinement is similar to that performed in silence removal module. In this stage (music removal), 19 MFCC features and their first- and second-order derivatives concatenated with ZCR have been used. Also, the STE has not exploited within the iterative classification process, and by its elimination, the speech with background music which has been classified as music has been changed to speech class.

2.3 Speaker segmentation

Growing window based on the delta Bayesian Information Criteria (ΔBIC) distance has been used as a speaker segmentation algorithm. It consists to make a research of a single change point in each frame of the audio recording. This research restarts from the next frame each time when a single change point is detected. In this case, the window size is initialized to 5 s, and for that frame, the distance ΔBIC is calculated. Indeed, a change point is declared as maximum point if the maxima in the window exceed a threshold value θ. In contrast, if there is no change point is detected, then the window size is increased by 2 s and the process is repeated till a change point is detected. We have to remember here that we deal only with speech frames as those no-speech are discarded by the SAD module. Thus, the corresponding locations of change points in the original audio found in these speech frames are declared as change points.

According to both broadcast diarization toolkits in [38, 39], speaker segmentation is performed in two phases: In the first one, a threshold value of zero is used by the ΔBIC-based change detection, and in the second one, the consecutive segments are merged when the ΔBIC score is positive. So, we can sum up these two phases in only one phase by considering maxima, which are greater than a threshold θ. By this way, we can reduce significantly the over segmentation engendered by the zero threshold ΔBIC-based segmentation. The ΔBIC expression is given as follows:

$$^{\triangle}\mathrm{BIC}(x_i) = \mathrm{Nlog}|\Sigma| - N_1 \log|\Sigma_1| - N_2 \log|\Sigma_2| - \frac{\lambda}{2}\left(d + \frac{d}{2}(d+1)\right)\mathrm{logN} \tag{1}$$

where Σ is the covariance matrix of the merged cluster (c_1 and c_2), Σ_1 of cluster c_1, and Σ_2 of cluster c_2, and N_1 and N_2 are, respectively, the number of acoustic frames in cluster c_1 and c_2, λ is a tunable parameter dependent on the data. $N = N_1 + N_2$ denotes the size of two merged clusters. In this speaker segmentation stage, only the 19 MFCC features have been used with their short time energies.

2.4 I-vector extraction

The success of I-vectors has been reached the language recognition [40, 41], and it is not only dedicated to speaker diarization, clustering tasks, and speaker recognition [42] ([2]). For the I-vector extraction, it is defined as the mapping of high-dimensional space to low-dimensional one named total variability space. The mathematic expression of mapping the super vector X to an I-vector x is given as follows:

$$X = X_{UBM} + Tx \tag{2}$$

where X_{UBM} denotes the Universal Background Model (UBM) and T is the rectangular matrix called total variability matrix. In this work, UBM is a diagonal covariance GMM of size 512, and it is one-time computation. Indeed, obtaining GMM for a segment is done by mean adapting the UBM for the feature vectors of the concerned segment.

2.4.1 WCCN

The use of the within-class covariance (WCC) matrix to normalize data variances has become widely dispread in the speaker recognition field [41, 43]. The need to be normalized for I-vectors which differ from one application to another is due to its representation of a wide range of the speech variability. Here, within-class covariance normalization (WCCN) is set to accomplish this task by penalizing axes which have high intra-class variance by making data rotation using decomposition of the inverse of the WCC matrix. After the I-vectors normalization, the different EAs and the TLBO technique have been applied in order to regroup the extracted I-vectors into an optimal number of clusters.

2.5 Speaker clustering

2.5.1 EAs

Under EAs, we can find evolution strategies, programming strategies, genetic programming, and evolutionary programming. All of these algorithms share a common structure based on simulating the evolution of individual structures through the process of selection, mutation, and reproduction. This process relies on the perceived performance of the individual structures as defined by the problem. The EAs start at first by initializing the population of candidate solutions, and then, new populations are created by applying reproduction operators (mutation and/or crossover). After that, the fitness evaluation of the resulting solutions is performed and the suitable selection strategy is applied in order to determine which are the solutions that will be maintained into the next solution. The iteration of the EAs process is performed as it is illustrated in the Fig. 1.

The algorithm of the EAs is given as follows:

1-Generation of an initial population P_0 of m individuals.

2- Set generational counter $k = 1$.

3-Evaluation of P_0 for fitness.

4- Beginning of an iterative process up to number of generations or termination criteria is reached.

4-a -Selection of parents such as: $P_{par} \longleftarrow P_{k-1}$

4-b- Getting offspring P_{offsp} by recombining parents.

4-c -Mutate of some offspring

4-d -Selection of population in order to survive into next generation

$P_k \longleftarrow P_{k-1} \cup P_{offsp}$.

4-e- Iteration of the generational counter such as: $k = k + 1$

Outline of the Evolutionary algorithm (EA)

2.5.2 DE algorithm

This algorithm uses non-linear and non-differentiable functions for optimization problems [44]. Indeed, differential evolution (DE) algorithm looks to optimize these functions from a set of randomly generated solutions using specific operators of recombination, selection, and replacement. The different steps of the DE algorithm are given below.

1- *Generation of an initial population*
2- *Calculation of the fitness function for each element of the population*
3- *Repeat*
4- *For each agent* $X_{i,G}$ *in the population do*
5- *Selection of the parent vectors* $X_{r1,G}$, $X_{r2,G}$ *and* $X_{r3,G}$ *where* $r_1 \# r_2 \# r_3 \# i$, $i = 1, \ldots, population\ size, G$ *is a partition*, $r_1 r_2 r_3 \in \{1, \ldots, population\ size\}$
6- *Generation of the perturbed vector* $V_{i,G+1}$ *according to the following equation:* $V_{i,G+1} = X_{r1,G} + F * (V_{r2,G-r3,G})$, *where* $F \in [0\ 1]$ *is a parameter used to control the amplitude of the differential variation at the time of disturbing the vector*
7- *Building the crossover vector* $U_{i\,G+1}$ *according to the following equation:*

$$U_{i,G+1} = \begin{cases} V_{i,G+1} & if\ rand \le cr\ \ or\ \ j = rnbr(i) \\ X_{i,G} & otherwise \end{cases}$$

where j=1…Dimension, crossover rate (cr) is a parameter to control the crossover operation and has to be determined by the user, rnbr(i) is chosen randomly at the time of the crossover process, and rand is the jth evaluation of a uniform random number generator with the outcome ϵ *[0:1].*

8- *Getting a new element for the population* $X_{i,G+1}$
9- *End for each*
10- *Until (Stop criteria are reaches)*

Pseudo code of the Differential Evolution (DE) algorithm

Concerning the PSO algorithm, more details about it can be found in [45].

2.5.3 TLBO algorithm

The TLBO method is one of the population-based methods, which relies on population of solutions to

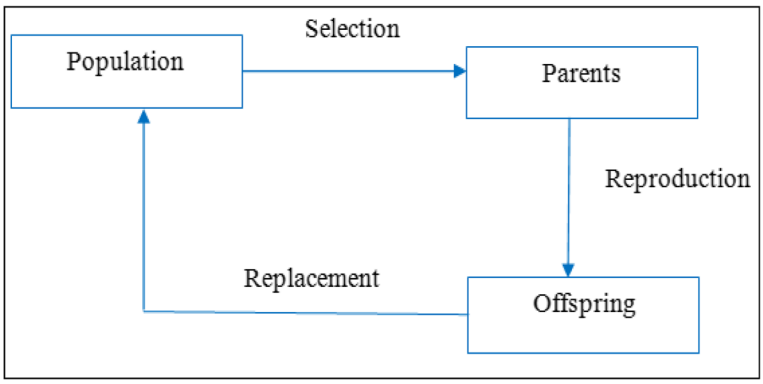

Fig. 1 Flow chart of EAs

reach the global one (solution). It has been used for clustering tasks [46]. The main idea behind this optimization method is to profit from the influence of a teacher on the learners' output in a class [47]. For this, in this algorithm, the population is considered as a group of learners. Concerning the optimization algorithms, the population is composed of different design variables, while for the TLBO approach, different design variables are similar to different subjects that are suggested to learners. Thus, concerning the learners' result here, it is similar to the "fitness" like in other population-based optimization techniques. In TLBO algorithm, the best solution obtained so far is considered to be given by the teacher.

The TLBO technique consists of two phases: the first one is the "teacher phase" and the second one is the "learner phase." Concerning the teacher phase, it consists to make learning from the teacher, and concerning the learner phase, the learning is made via the interaction between learners.

2.5.3.1 Teacher phase The main idea behind this phase is to consider a teacher as the knowledgeable person in the society who transfers his knowledge among learners, which can contribute to increase the knowledge level of the entire class and allows learners to get good marks or grades. So, the mean of the class is increased by the teacher's capability, i.e., moving the mean M1 towards the teacher's level is performed according to the capability of the teacher T1, which enables to increase the learner's level into a new mean M2. Also, the student's knowledge is increased according to his quality in the class and to the teaching quality given by the teacher T1. Changing the student's quality from M1 to M2 is relied on the effort of the teacher T1. Consequently, the student at the new level needs a new teacher T2 who has more quality than him [45].

Let us consider M_i the mean and T_i the teacher at any iteration. Trying to move M_i by the teacher T_i towards its own level engenders consequently the creation of M_{new}, which is a design of the new mean T_i. The solution update is performed according to the difference between the existing and new mean, and it is given by the following expression:

$$\text{Difference}_{\text{mean}_i} = r_i(M_{\text{new}} - T_F M_i) \tag{3}$$

where T_F is the teaching factor which is responsible for taking the decision about the mean value to change, and r_i is the random number in the range [0 1]. Also, the value of T_F can be either 1 or 2, which is again a heuristic step or it is decided randomly with equal probability as:

$$T_F = \text{round}[1 + \text{rand}(0,1) \times (2-1)] \tag{4}$$

This subtraction modifies the existing solution, and it depends on the following expression:

$$X_{\text{new},i} = X_{\text{old},i} + \text{Difference}_{\text{mean}_i} \tag{5}$$

2.5.3.2 Learner phase Increasing the learners' knowledge is performed from the teacher through an input and via the interaction between learners themselves. Each learner has a randomization interaction with other learners with the assistance of group discussions, presentations, formal communication, and others. Each time that the learner's knowledge is less than the knowledge of another one, then the learner will learn something new [45]. Thus, the modification in the learner is given by the following algorithm:

for $i = 1{:}P_n$

Randomly select two learners X_i *and* X_j, *where* $i \neq j$

if $f(X_i) < f(X_j)$

$$X_{new,i} = X_{old,i} + r_i(X_i - X_j),$$

Where, $f(X_i)$ *and* $f(X_i)$ *are respectively the fitness value of the learner* X_i *and* X_j.

Else

$$X_{new,i} = X_{old,i} + r_i(X_j - X_i)$$

end if

end for

If X_{new} gives a better function value then it is accepted.

2.6 Statistical clustering criteria

The measure of the partition's adequacy is performed by different statistical criteria, which allow a comparison through different partitions. Other transformations can be usually involved by these criteria, such as the trace or determinant of both pooled-within groups scatter matrix (**W**) and between groups scatter matrix (**B**). The pooled-within scatter matrix (**W**) is expressed as follows:

$$W = \sum_{k=1}^{g} W_k \tag{6}$$

where W_k denotes the variance matrix of the objects' features allocated to cluster $C_k (k = 1, \ldots, g)$. Therefore, if $X_{l^{(k)}}$ designs the lth object in cluster C_k and n_k, the number of objects in cluster C_k, then:

$$W_k = \sum_{l=1}^{n_k} n_k \left(x_l^{(k)} - \overline{x}^{(k)}\right)\left(x_l^{(k)} - \overline{x}^{(k)}\right)' \quad (7)$$

Where, $\overline{x}^{(k)} = \left(\sum_{l=1}^{n_k} x_{l^{(k)}}\right)/n_k$ is the vector of the centroids for cluster C_k [48]. In this work, we have used the trace of the pooled-within groups scatter matrix (W) as a distance measure of the fitness function and it is denoted by WCD. Also, the computation of the fitness function has been carried out according to distance measures using DB and CS indexes as clustering validity index.

2.6.1 DB index

The minimization of the average similarity between each cluster and the one most similar to it is performed by this clustering validity index, which is defined as [49]:

$$\mathrm{DB} = \frac{1}{K}\sum_{k=1}^{K} \max\left(\frac{\mathrm{diam}(C_k) + \mathrm{diam}(C_{kk})}{\mathrm{dist}(C_k, C_{\mathrm{kk}})}\right) \quad (8)$$

with $kk = 1, \ldots, K$ and $k \neq kk$.

Where, diam denotes the perfect diameter which is defined as the inter-cluster and intra-cluster distance of C_k and C_{kk} clusters.

2.6.2 CS index

The Constructability Score (CS) Index measures the particle's fitness, and it is defined such as [50]:

$$\mathrm{CS}(K) = \frac{\frac{1}{T}\sum_{i=1}^{T}\left\{\frac{1}{N_i}\sum_{x_j \in C_i} \max_{x_k \in C_i}\left\{d(x_j, x_k)\right\}\right\}}{\frac{1}{T}\sum_{i-1}^{T}\left\{\min_{j \in T, j \neq i}\left\{d(Z_i, Z_j)\right\}\right.} = \frac{\sum_{i-1}^{T}\left\{\frac{1}{N_i}\sum_{x_j \in C_i} \max_{x_k \in C_i}\left\{d(x_j, x_k)\right\}\right\}}{\sum_{i=1}^{T}\left\{\min_{j \in T, j \neq i}\left\{d(Z_i, Z_j)\right\}\right.} \quad (9)$$

$$Z_i = \frac{1}{N_i}\sum_{x_j \in C_i} x_j, i = 1, 2, \ldots T; \quad d\left(x_j, x_k\right) = \sqrt{\sum_{p=1}^{N_d}\left(x_{\mathrm{jp}} - x_{\mathrm{kp}}\right)^2} \quad (10)$$

Where Z_i denotes the cluster center of C_i, C_i designs the set whose elements are the data points attributed to the ith cluster, N_i the number of elements in C_i, and d designs a distance function.

The CS measure is also a function of the ratio of the sum of within-cluster scatter between-cluster separation [45]. In order to reach proper clustering results for the PSO algorithm, this measure (CS measure) has to be minimized. Consequently, the computation of the fitness function for each individual particle is expressed as follows:

$$F = \frac{1}{\mathrm{CS}_i + \mathrm{eps}} \quad (11)$$

where CS_i is the CS measure computed for the ith particle, and eps is a very small-valued constant.

3 Experiments and analysis

3.1 Evaluation criteria

The evaluation of speaker diarization is an optimal measure obtained by mapping one-to-one of the reference speakers' identities (IDs) and the hypothesis ones. The first metric for this task is concerned with the speaker match error, which corresponds to the fraction of speaker's time, which is attributed incorrectly to the correct speaker, obtaining consequently the optimum speaker mapping. The second metric is the overall speaker diarization error rate (DER), which involves the missed and false alarm speaker times. This metric is defined in absence of overlapping such as:

$$\mathrm{DER} = E1 + E2 + E3 \quad (12)$$

where $E1 = \frac{\text{missed speech time}}{S} \times 100$, $E2 = \frac{\text{false alarm speech time}}{S} \times 100$, and $E3 = \frac{\text{incorectelly labelled speech time}}{S} \times 100$

Where, s is the total speech time. For $E3$, it is engendered by errors in both speaker segmentation and clustering stages, and it is often named speaker error (SPK_ERR). The illustration of these measures is given in Fig. 2.

The performance analysis of the speaker clustering methods involves also the average frame-level cluster purity as well as the cluster coverage [51]. For the speaker purity performance, it is calculated as the number of frames by the dominant speaker in a cluster divided by the total number of frames in the cluster. Concerning the cluster coverage, it takes into consideration the dispersion of a given speaker data across clusters and it is given by the percentage of speaker's frames in cluster, which contain most of the speaker data [52]. The purity of a cluster p_i is given as follows:

$$p_i = \sum_{j=1}^{N_s} \frac{n_{\mathrm{ij}}^2}{n_i^2} \quad (13)$$

where n_{ij} is the total number of frames in cluster i spoken by speaker j, n_i is the total number of frames in cluster i, and N_s is the total number of speakers. The average cluster purity (acp) is defined as follows:

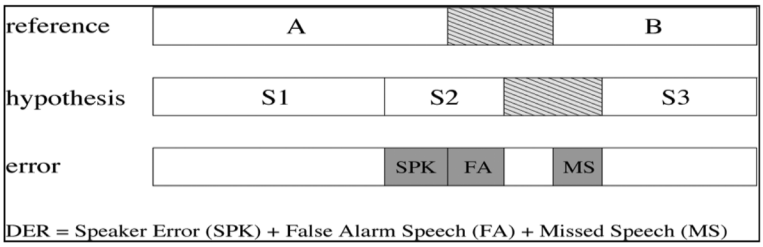

Fig. 2 Example of performance measures for speaker diarization task used in the NIST RT-04F evaluations

$$\text{acp} = \frac{1}{N}\sum_{i=1}^{N_c} p_i.n_i \tag{14}$$

Where N is the total number of frames for the speaker purity p_j and for the average speaker purity (asp), which are respectively defined as:

$$p_j = \sum_{i=1}^{N_c} \frac{n_{ij}^2}{n_j^2} \tag{15}$$

and

$$\text{asp} = \frac{1}{N}\sum_{j=1}^{N_s} p_j.n_j \tag{16}$$

Where n_j is the total number of frames spoken by speaker j and N_c is the total number of clusters.

We can mention here that good measure limitation of a speaker to belong to only one cluster is given by asp and good measure limitation of a cluster to be assigned to only one speaker is given by acp [52]. So, we have used an overall evaluation criterion which is the square of the product of these two factors such as:

$$K = \sqrt{\text{asp} \times \text{acp}} \tag{17}$$

It is important to mention that the DER values obtained in all experiments of this work are the overall diarization error rates which can be calculate as the averages of the individual DER per episode multiplied by the duration of the episode.

Also, the segments obtained after segmentation should contain a single speaker and give the correct speaker

Table 1 The best parameters setting given the best cost solution for all proposed algorithms

GA parameters	PSO parameters
Maximum number of iterations = 200	
Population size(nPop) = 100	Constriction coefficients
Crossover percentage = 0.7	phi1 = 2.05, phi2 = 2.05 phi = phi1 + phi2
Number of offsprings (nc) = 2*round(pc*nPop/2)	
Mutation percentage (pm) = 0.3	chi = 2/(phi-2 + sqrt(phi^2-4*phi))
Number of mutants (nm) = round (pm*nPop)	Inertia weight w = chi
Mutation rate (mu) = 0.02	Inertia weight damping ratio (wdamp) = 1
Selection pressure (beta) = 8	Personal learning coefficient (c1) = chi*phi1
Gamma = 0.2	Global learning coefficient (c2) = chi*phi2
	Velocity maximal = 0.1*(VarMax-VarMin)
	Velocity minimal = − VelMax
	VarMin = − 10; VarMax = 10
DE parameters	
Maximum number of iterations (MaxIt) = 200	
Population size (nPop) = 50	
Lower bound of scaling factor (beta_min) $C_r$min = 0.2	
Upper bound of scaling factor (beta_max) $C_r$max = 0.8	
Crossover probability (pCR) = 0.2	
TLBO parameters	
MaxIt = 1000; nPop = 50; $T_F = 1$	

turns at their boundaries. In fact, there are many kind of errors attached to speaker turns detection which can be recognized. In our work, we have used PRC, RCL, and F as assessment measures. For the first two, they are defined as:

$$\mathrm{RCL} = \frac{\text{number of correctly found turns}}{\text{total number of correct turns}} \tag{18}$$

$$\mathrm{PRC} = \frac{\text{number of correctly found turns}}{\text{total number of turns found}} \tag{19}$$

In the purpose of evaluation of the segmentation quality, F has been used as a measure combination of RCL and PRC of change detection. Thus, F is defined such as:

$$F = \frac{2 \times \mathrm{RCL} \times \mathrm{PRC}}{\mathrm{RCL} + \mathrm{PRC}} \tag{20}$$

3.2 NDTV evaluation corpus

The experiments presented below for speaker diarization have been developed on MATLAB and have been tested on the News database (NDTV). The development database (NDTV) contains 22 episodes of the Hindu news Headlines Now Show from the NDTV news channel. It includes English new reading of a length of 4 h and 15 min with Indian accent, and it was manually annotating. The dominant speaker in the episodes is the anchor as he takes more much time talking than other speakers. Also, across all episodes, the anchors differentiate to each another. The announcement of the headlines is accompanied with music in the background, which is a common point in all episodes. In addition, the speaker in a single episode is labeled by its genre, background environment (clean, noise, or music), and identity (ID). Therefore, the silence segment length varies from 1 to 5 s, and there is no advertisement jingles presented in the dataset. For the silence, noise, speaker's pauses, or music, they are labeled as no-speech, which represents 7% of the total recording. Thus, the annotation of the speaker overlap has been performed with the most dominant speaker in the overlap.

Table 2 SAD results obtained using: silence removal module alone, music removal module alone, and by cascading both modules

Error method	MSR	FASR	Total SAD
Silence removal	1.37	3.31	4.68
Music removal	1.42	5.62	7.04
Cascade	2.79	8.93	11.72

Table 3 DER results obtained using ILP clustering algorithm

	BIC criterion					
	$\lambda = 1$ $\Theta = 0$	$\lambda = 1$ $\Theta = 1000$	$\lambda = 1$ $\Theta = 2000$	$\lambda = 10$ $\Theta = 0$	$\lambda = 10$ $\Theta = 1000$	$\lambda = 10$ $\Theta = 2000$
ILP	31.52	23.67	12.35	33.41	16.54	16.10

3.3 Implementation and parameter setting

The parameter setting, which has given the best cost solution for all implemented algorithms, is given in Table 1.

3.4 Results

The Speech Activity Detector (SAD) has been implemented by cascading the silence removal module to the music removal module. This implementation has been shown an improvement in both missed speech ratio (MSR) and false alarm speech ratio (FASR) comparing to the implementation of each module alone. Indeed, the implementation of the silence removal module alone or the music removal module alone engenders high false alarm rate. The results obtained for both cases are summarized in Table 2.

Beside the evolutionary and TLBO algorithms, our model has been tested with the well competitive algorithm in speaker diarization, which is the Integrated Linear Programming (ILP) algorithm. Table 3 exhibits the different DER values obtained with ILP algorithm for different Bayesian Information Criteria (BIC) parameters. Here, the best DER values have been reached with high λ and θ values. In contrast, the low values of these parameters have led to high DER values. Thus, the latter parameter setting engenders an over segmentation, which due to the increase of the average duration of segments, which is caused by increasing the θ value.

Also, our model has been tested with hierarchical agglomerative clustering (HAC) and ILP algorithms using GMM and I-vectors as speaker models. The best results in this test have been obtained with ILP-I-vector clustering as it is mentioned in Table 4. This proves the superiority of ILP clustering compared to HAC method.

Therefore, our model has been tested using different evolutionary algorithms (EAs) with specificity for GA algorithm for which we have made different variation in its control parameters such as the selection and crossover, as well as the clustering validity index. The

Table 4 Best DER result obtained for both speaker models and clustering algorithms

	Clustering algorithm	
Speaker model	HAC	ILP
GMM	19.45	17.15
I-vectors	16.95	*16.10*

Table 5 DER results for GA algorithm obtained using DB and CD clustering validity indexes, with different selection, and the DER results obtained with "sphere" cost function for both GA-based binary representation and GA-based real-coded representation

			ACP %	ASP %	DER %
GA-based binary representation	Random selection		93.54	98.13	14.62
	Roulette wheel selection		92.72	88.14	14.8
	Tournament selection		90.37	86.16	14.4
Ga-based real-coded represention	Roulette wheel selection		91.32	87.63	14.3
	Tournament selection		90.17	85.90	14.19
	Random selection		89.35	85.75	14.35
GA with DB and CD indexes	Roulette wheel selection	DB index	94.36	91.60	14.19
		CD index	95.17	92.85	14.12

computational cost in this section is given by the objective function, which has reached the best cost solution. From Tables 5 and 6, we can mention that the CD index is the best clustering validity index, which has reached the best results for GA algorithm in terms of ACP, ASP, and DER comparing to those obtained with DB and WCD indexes. This is due to the best cost solution reached by this index comparing to other ones. In addition, the GA algorithm with CD index has exceeded in the achieved results of both GA-based binary representation and GA-based real-coded representation using "sphere" as a cost function. From Table 6, we can also show that the *single-point crossover* and the *double-point crossover* are the best *crossover* modes, which have led to good results for GA algorithm using WCD index. For the *selection*, in some cases, the *Roulette wheel selection* seems the best kind of selection, and in other ones, the *tournament selection* is the best one.

Also, the DE algorithm is the best EAs in terms of DER, ACP, and ASP results in which it contributes to obtain the lowest values, and this is in virtue of the CD index. But, comparing DE algorithm to TLBO technique, the best DER, ACP, and ASP values have been obtained by the TLBO technique as it is mentioned in Table 7.

From the Fig. 3b, we can show the domination of the TLBO algorithm compared to other algorithms in terms of indexing results (K) in which it has reached the best value (97.12%). In addition, for the same evaluation, the CD index is better than DB index using both GA and DE algorithms (Fig. 3a) and its best result has been achieved with DE algorithm (95.12%). Therefore, for different selection and crossover combinations using GA algorithm, the Roulette wheel selection used with double-point crossover is the best combination, which has succeed to reach the best indexing results (K) (91.25%) (Fig. 3c). Also, we have to mention here that the different indexing results (K) have been obtained with different WAV files, which contain a number of speakers ranged between three and five speakers.

Concerning the segmentation results (F), our system has been evaluated using GA algorithm with DB index, DE algorithm with CD index, PSO algorithm, and TLBO algorithm. This evaluation has been performed on WAV files, which contain between three and five speakers. As it is shown in Table 8, the TLBO algorithm remains the best in terms of segmentation results compared to other algorithms. Indeed, it has reached the best average segmentation scores (F) with the WAV files, which contain either three or five speakers (98.45 and 97.84, respectively). Also, we can see clearly here that increasing the number of speakers in the audio files decreasing consequently the segmentation results. In fact, the record in terms of best results reached by the TLBO algorithm has been achieved in virtue of SAD which has contributed sharply to decrease the percentage of both missed and false speech alarms.

To look for the efficiency of our proposed system, we have tested it on two datasets of News Broadcast shows,

Table 6 DER results obtained for GA algorithm with WCD index, using different selection as well as different crossover modes

Crossover/selection type	Roulette wheel selection	Tournament selection	Random selection
Uniform crossover	ACP 92.44 ASP 88.73 DER 14.52	ACP 92.41 ASP 88.53 DER 14.5	ACP 91.62 ASP 88.98 DER 14.67
Single-point crossover	ACP 91 87 ASP 88.66 DER 14.25	ACP 90.65 ASP 87.63 DER 14.38	ACP 89.52 ASP 87.14 DER 14.43
Double-point crossover	ACP 92.27 ASP 90.26 DER 14.22	ACP 91.35 ASP 88.87 DER 14.35	ACP 91.83 ASP 89.43 DER 14.32

Table 7 ASP, ACP, and DER results obtained using PSO and DE algorithms with different selection modes and with different clustering validity indexes (DB and CS indexes). Also, ASP, ACP, and DER results are obtained using TLBO algorithm

			ACP %	ASP %	DER %
PSO algorithm	Roulette wheel selection		93.54	89.13	14.62
	Tournament selection		92.72	88.14	14.8
	Random selection		90.37	86.16	14.4
DE algorithm	Roulette wheel selection		91.32	87.63	14.3
	Tournament selection		91.87	88.66	14.25
	Random selection		90.54	88.48	14.37
	Roulette wheel selection	DB index	95.35	91.35	13.87
		CD index	96.88	93.40	13.72
TLBO algorithm			98.63	95.65	13.27

which are RT-04F and ESTER datasets. We have performed a comparison between the best proposed algorithm in this work, which is the TLBO algorithm and the multi-stage portioning system proposed in [53]. This system used BIC-based agglomerative clustering (AC) followed by another clustering stage of GMM-based speaker identification as well as a post-processing stage. Indeed, the proposed system in [53] is composed of baseline portioning system (*c-std)*, speaker identification system (*c-sid)* (with threshold δ), and agglomerative

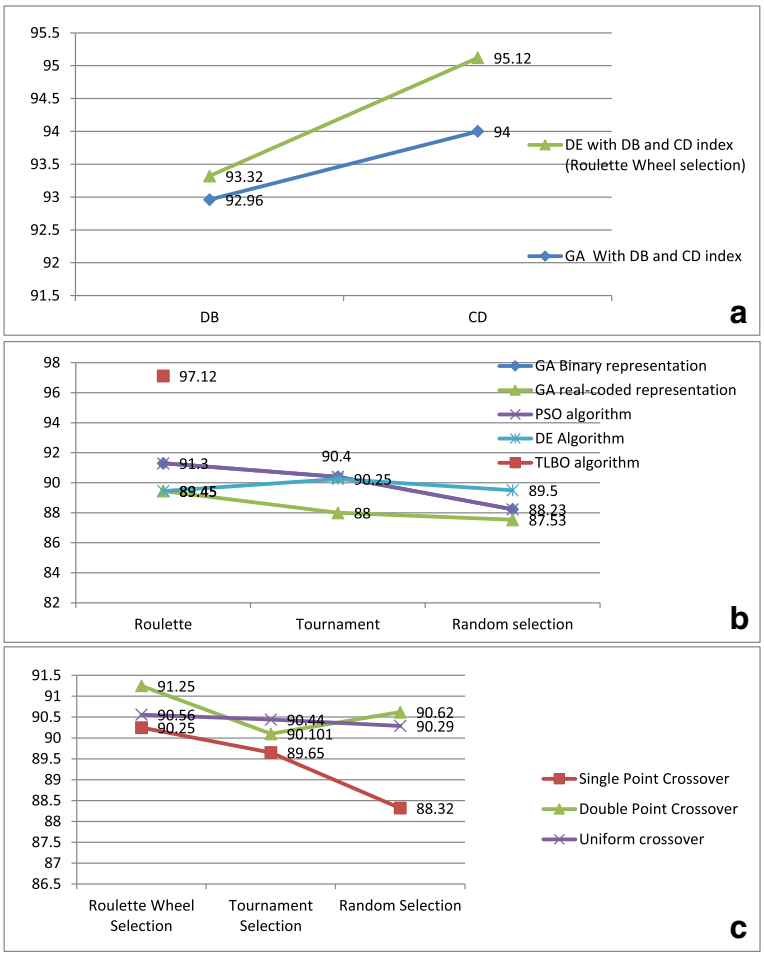

Fig. 3 a Indexing results (K) obtained using GA and DE algorithms with DB and CD indexes. **b** Indexing results (K) obtained with GA, PSO, DE, and TLBO algorithms. **c** Indexing results (K) obtained using GA algorithm with different crossover modes

Table 8 Segmentation results using NDTV dataset

Number of speakers	File duration	F			
		GA	PSO	DE	TLBO
3	10 mn	96.4	97.1	97.3	97.62
	10 mn and 2 s	96.75	97.12	97.42	97.84
	10 mn and 33 s	97.3	97.85	98.32	*98.45*
	Average	96.81	97.35	97.54	97.97
5	10 mn	95.52	97.32	97.12	97.54
	10 mn and 2 s	95.95	97.32	97.26	97.69
	10 mn and 33 s	95.3	97.65	98.00	98.30
	Average	95.59	97.26	97.46	*97.84*

clustering system based on BIC (*c-bic)* as well as an automatic speech recognition system with post-processing (*p-asr)*, which has been proposed in [54]. As it is showed in Table 9, we can say that the TLBO technique has been succeeded to reach competitive performance results on both datasets compared to those algorithms used in [53]. Indeed, on the *dev1* dataset, the *c-sid* system has reached the best overall DER value (7.1%) compared to the TLBO algorithm (7.249%). Also, this system (with also *p-asr system*) has exceeded the TLBO algorithm on the *dev2* dataset in terms of overall DER result in which it has reached the best value (7.6%). In addition, the best overall DER value (11.5%) has been achieved by *c-sid* system (with a threshold δ = 1.5) on the ESTER dataset against 12.3% for the TLBO algorithm. Therefore, using post-evaluation on ESTER dataset, the *c-sid* system (δ = 2.0) has succeeded to reach good overall DER result (9.1%). We can mention from Table 9 that the speaker errors (SPK) have been increased by increasing the number of speakers in the audio files as it is clearly demonstrated by the high SPK values (12.2 and 11.5%) obtained with ABC and NBC audio files, respectively. Consequently, the high SPK values contribute sharply to obtain high overall DER values. Concerning the missed speech (MS) values, they are so low in all tests performed on both RT-04F and ESTER datasets, while the false alarm (FA) values obtained on the same datasets are quiet high.

Table 9 Performance results of TLBO algorithm, *c-bic*, *c-sid,* and *p-asr* systems obtained on the RT-04F and ESTER datasets. Scores are given for missed speech (MS), false alarms (FA), speaker errors (SPK), and overall diarization error rate (DER). #REF and #Sys are, respectively, the *reference* and *system* speaker number

RT-04F dev1 dataset							
System	Method	#Ref	#Sys	MS	FA	SPK	Overall DER
Dev1	c-sid	121	161	0.4	1.3	5.4	*7.1*
	TLBO algorithm	121	161	0.383	1.116	5.75	*7.249*
Show	ABC	27	35	1.4	1.1	12.2	14.7
	VOA	20	22	0.2	1.1	2.1	3.4
	PRI	27	29	0.1	0.8	2.7	3.6
	NBC	21	30	0.1	0.9	11.5	12.5
	CNN	*16*	*19*	*0.4*	*1.2*	*5.4*	*7.0*
	MNB	10	13	0.1	1.6	0.6	2.3
Dev2	c-sid	90	130	0.5	3.1	4.1	*7.6*
	TLBO algorithm	90	130	0.516	3.083	4.216	*7.725*
Show	CSPN	3	4	0.2	2.8	0.1	3.1
	CNN	17	20	0.6	4.1	4.9	9.6
	PBS	27	28	0.1	2.6	7.2	10.0
	ABC	23	26	2.1	6.7	12.1	20.9
	CNNHL	9	15	0.0	1.4	0.3	1.7
	CNBC	11	16	0.1	0.9	0.7	1.7
RT-04F dev2 dataset							
c-bic	–	–		0.4	1.8	14.8	*17.0*
c-sid (δ = 0.1)	–	–		0.4	1.8	6.9	*9.1*
p-asr	–	–		0.6	1.1	5.2	*7.6*
TLBO algorithm	–	–		0.6	1.8	7.8	*10.2*
ESTER development dataset							
c-bic	–	–	0.7	1.0		12.1	13.8
c-sid (δ = 1.5)	–	–	0.7	1.0		9.8	*11.5*
TLBO algorithm	–	–	0.6	1.0		9.7	*12.3*
Post-evaluation result on ESTER dataset							
c-sid (δ = *2.0*)	–	–	0.7	1.0		7.4	*9.1*

4 Conclusions

In this paper, we have used the EAs and teaching-learning-based optimization technique (TLBO) in the speaker clustering stage for speaker diarization of broadcast news. We have evaluated the proposed model on NDTV database which consists of different speakers. The results have demonstrated the high performance of the TLBO algorithm in terms of ASP, ACP, and DER results comparing to different EAs using different clustering validity indexes (CD, WCD, and DB indexes) and to ILP algorithm. Future work may consist of more improving of the evaluated performances by making hybridization between TLBO technique and EAs with k-means algorithm (Table 10).

Table 10 Benchmark function

Function	Formula	Range	Optima
Sphere	$F_1(x) = \sum_{i=1}^{D} x_i^2$	[– 100, 100]	0

Authors' contributions

DK and HS designed the speaker diarization model, performed the experimental evaluation, and drafted the manuscript. CA reviewed the paper and provided some advice. All authors read and approved the final manuscript.

Competing interests

The authors declare that they have no competing interests.

Publisher's Note

Author details

[1]Research Unity of Analysis and Processing of Electrical and Energetic systems, Faculty of Sciences of Tunis, University of Tunis El Manar, 2092 Tunis, Tunisia. [2]Research Laboratory of Information and Image Processing, National School of Engineers of Tunis, University of Tunis El Manar, 2092 Tunis, Tunisia.

Received: 15 March 2017 Accepted: 30 August 2017
Published online: 19 September 2017

References

1. J. Kennedy, Some issues and practices for particle swarms, in IEEE Swarm Intelligence Symposium, pp.162-169, 2007.
2. A. Veiga, C. Lopes, and F. Perdig~ao. *Speaker diarization using Gaussian mixture turns and segment matching*. Proc. FALA, 2010.
3. Tranter, S., & Reynolds, D. (2006). An overview of automatic speaker diarization systems, Audio, Speech, and Language Processing. *IEEE Transactions on, 14*(5), 1557–1565.
4. Gauvain, J. L., Lamel, L., & Adda, G. (1998). Partitioning and transcription of broadcast news data. In *ICSLP* (Vol. 98-5, pp. 1335–1338).
5. Tang, H., Chu, S., et al. (2012). Partially supervised speaker clustering. *IEEE Trans. Pattern Anal. Mach. Intell., 34*, 959–971.
6. Li, Y.-X., Wu, Y., & He, Q.-H. (2012). Feature mean distance based speaker clustering for short speech segments. *Journal of Electronics & Information Technology, 34*, 1404–1407 (In Chinese).
7. W. Jeon, C. Ma, D. Macho, An utterance comparison model for speaker clustering using factor analysis, IEEE International Conference on Acoustics, Speech and Signal Processing, 2011, pp. 4528-4531.
8. Iso, K. (2010). *Speaker clustering using vector quantization and spectral clustering* (pp. 4986–4989). Dallas: IEEE International Conference on Acoustics, Speech and Signal Processing.
9. Ning, H. Z., Liu, M., Tang, H., et al. (2006). *A spectral clustering approach to speaker diarization* (pp. 2178–2181). Pittsburgh: IEEE Proceedings of the 9th International Conference on Spoken Language Processing.
10. Wu, K., Song, Y., Guo, W., & Dai, L. (2012). Intra-conversation intra-speaker variability compensation for speaker clustering. In *Chinese Spoken Language Processing (ISCSLP), 2012 8th International Symposium on* (pp. 330–334). https://doi.org/10.1109/ISCSLP.2012.6423465.
11. Rouvier M., Favre B. Speaker adaptation of DNN-based ASR with i-vectors: does it actually adapt models to speakers? *INTER SPEECH* 14-18 September 2014, Singapore.
12. Wei-Ho Tsai and Hsin-Min Wang, Speaker clustering based on minimum rand index, Department of Electronic Engineering, National Taipei University of Technology, Taipei, Taiwan, 2009.
13. S. Paterlini, T. Krink *Differential evolution and particle swarm optimization in partitional clustering*. Science direct. Computational Statistics & Data Analysis 50 (2006) 1220 – 1247.
14. Goldberg, Genetic algorithms in search, optimization, and machine learning, Addison-Wesley Longman Publishing Co., Inc. Boston, MA, USA ©1989.
15. G. Liu, Y. Xiang Li and G. *He* Design of digital FIR filters using differential evolution algorithm based on reserved genes. 978-1-4244-8126-2/10/$26.00 ©2010 IEEE.
16. Jain, A.K., Murty, M.N. and Flynn, P.J. (1999), Data clustering: a review. *ACM Computing Surveys*, 31.264-323. https://doi.org/10.1145/331499.331504.
17. Kennedy, J., & Eberhart, R. C. (1995). Particle swarm optimization. In *Proceedings of IEEE International Conference on Neural Networks, Piscataway, New Jersey* (pp. 1942–1948).
18. Clerc, M., & Kennedy, J. (2002). The particle swarm-–explosion, stability, and convergence in a multidimensional complex space. *IEEE Trans. Evol. Comput., 6*(1), 58–73.
19. Tillett, J. C., Rao, R. M., Sahin, F., & Rao, T. M. (2003). Particle swarm optimization for clustering of wireless sensors. In *Proceedings of Society of Photo-Optical Instrumentation Engineers* (Vol. 5100, p. No. 73).
20. Cui, X., Palathingal, P., & Potok, T. E. (2005). Document clustering using particle swarm optimization. In *IEEE Swarm Intelligence Symposium, Pasadena, California* (pp. 185–191).
21. Alireza, Ahmadyfard and Hamidreza Modares, Combining PSO and k-means to enhance data clustering, 2008 International Symposium on Telecommunications, pp. 688-691, 2008.
22. S.M. Mirrezaie & S.M. Ahadi, Speaker diarization in a multi-speaker environment using particle swarm optimization and mutual information, Department of Electrical Engineering, Amirkabir University of Technology 424 Hafez Avenue, Tehran 15914, Iran. 978-1-4244-2571-6/08/$25.00 ©2008 IEEE.
23. M. Zhang, W. Zhang and Y. Sun, Chaotic co-evolutionary algorithm based on differential evolution and particle swarm optimization, Proceedings of the IEEE International Conference on Automation and Logistics Shenyang, China August 2009.
24. R. Yadav, D. Mandal, Optimization of artificial neural network for speaker recognition using particle swarm optimization. International Journal of Soft Computing and Engineering (IJSCE) ISSN: 2231-2307, Volume-1, Issue-3, July 2011.
25. Poli, R., Kennedy, J., & Blackwell, T. (2007). Particle swarm optimization. *Swarm Intelligence, 1*(1), 33–57.
26. R. Poli, An analysis of publications on particle swarm optimization applications, Essex, UK: Department of Computer Science, University of Essex, May - Nov2007.
27. Sun, J., Feng, B., & Xu, W. (2004). Particle swarm optimization with particles having quantum behavior. In *Proceedings of Congress on Evolutionary Computation, Portland (OR, USA)* (pp. 325–331).
28. Z. Hong, Z. JianHua Application of differential evolution optimization based Gaussian mixture models to speaker recognition. School of Information Science and Engineering, East China University of Science and Technology, Shanghai 200237. 978-1-4799-3708-0/14/$31.00_c , 2014, IEEE.
29. R. Storn and K. V. Price, Differential evolution: a simple and efficient adaptive scheme for global optimization over continuous spaces, ICSI, USA, Tech. Rep. TR-95-012, 1995 [Online]. Available: http://www.icsi.berkeley.edu/~storn/litera.html.
30. R.V. Rao and V. Patel, An elitist teaching-learning-based optimization algorithms for solving complex constrained optimization problems. International Journal of Industrial Engineering Computations, vol. 3, no. 4, pp. 535–560, 2012.
31. S. O. Degertekin and M. S. Hayalioglu, Sizing truss structures using teaching-learning-based optimization. Computers and Structures, vol. 119, pp. 177–188, 2013.
32. R. V. Rao and V. Patel, An improved teaching-learning-based optimization algorithm for solving unconstrained optimization problems. Scientia Iranica, vol. 20, no. 3, pp. 710–720, 2013.
33. T. Niknam, F. Golestaneh, and M. S. Sadeghi, -Multiobjective teaching-learning-based optimization for dynamic economic emission dispatch. IEEE Systems Journal, vol. 6, no. 2, pp. 341– 352, 2012.
34. Zou, F., Wang, L., Hei, X., Chen, D., & Yang, D. (2014). Teaching-learning-based optimization with dynamic group strategy for global optimization. *Inf. Sci., 273*, 112–131.
35. Wang, L., Zou, F., Hei, X., Yang, D., Chen, D., & Jiang, Q. (2014). An improved teaching learning-based optimization with neighborhood search for applications of ANN. *Neurocomputing, 143*, 231–247.
36. Suresh Chandra Satapathy, Anima Naik and K Parvathi, A teaching learning based optimization based on orthogonal design for solving global optimization problems. SpringerPlus 2013.
37. Waghmare, G. (2013). Comments on a note on teaching-learning-based optimization algorithm. *Information Sciences, 229*, 159–169.
38. M. Rouvier, G. Dupuy, P. Gay, E. Khoury, T. Merlin, and S. Meignier, An open source state-of-the-art toolbox for broadcast news diarization. Technical report, Idiap, 2013.
39. M. Anthonius, H. Huijbregts, Segmentation, diarization, and speech transcription, surprise data unraveled. 2008.
40. X. Anguera and J. Hernando, Xbic, Real-time cross probabilities measure for speaker segmentation. Univ. California Berkeley, ICSIBerkeley Tech. Rep, 2005.

41. S. Cheng, H. Min Wang, and H. Fu, Bic-based speaker segmentation using divide-and-conquer strategies with application to speaker diarization. Audio, Speech, and Language Processing, IEEE Transactions on, 18(1):141-157, 2010.
42. T. Nguyen, H. Sun, S. Zhao, S. Khine, HD Tran, TLN Ma, B Ma, ES Chng, and H Li. The speaker diarization systems for RT 2009. In RT'09, NIST Rich Transcription Workshop, May 28-29, 2009, Melbourne, Florida, USA, volume 14, pages 1740, 2009.
43. H. K. Maganti, P. Motlicek, and D. Gatica-Perez. Unsupervised speech/non-speech detection for automatic speech recognition in meeting rooms. In Acoustics, Speech and Signal Processing, 2007. ICASSP 2007. IEEE international conference on, volume 4, pages IV{1037. IEEE, 2007.
44. Luz-Marina Sierra, Carlos Cobos, Juan-Carlos Corrales (2014), Continuous optimization based on a hybridization of differential evolution with K-means, In computer science, November, 2014.
45. Murty, M.R., *et al.* (2014), Automatic clustering using teaching learning based optimization. AppliedMathematics, 5, 1202-1211. https://doi.org/10.4236/am.2014.58111.
46. Pal, S.K. and Majumder, D.D. (1977). Fuzzy sets and decision making approaches in vowel and speaker recognition. IEEE Transactions on Systems, Man, and Cybernetics, **7**, 625-629.
47. Satapathy, S.C. and Naik, A. (2011), Data clustering based on teaching-learning-based optimization. *Lecture Notes in Computer Science*, 7077, 148-156.
48. Blake, C., Keough, E., & Merz, C. J. (1998). *UCI repository of machine learning database* http://www.ics.uci.edu/~mlearn/MLrepository.html.
49. D. Davies and D. Bouldin, A cluster separation measure. Determining the number of clusters In CROKI2 algorithm, IEEE PAMI, vol. 1, no. 2, pp. 224–227, 1979.
50. Malika Charrad RIADI and CEDRIC, Determining the number of clusters In CROKI2 algorithm. First Meeting on Statistics and Data Mining, MSriXM '09, 2009
51. S. Bozonnet, NWD Evans, and C. Fredouille, The LIA-EURECOM RT'09 speaker diarization system: enhancements in speaker Gaussian and cluster purification. In Acoustics Speech and Signal Processing (ICASSP), 2010 IEEE International Conference on, pages 4958{4961. IEEE, 2010.
52. S.M. Mirrezaie & S.M. Ahadi (2008), Speaker diarization in a multi-speaker environment using particle swarm optimization and mutual information. 978-1-4244-2571-6/08/$25.00 ©2008 IEEE.
53. C. Barras, X. Zhu, S. Meignier, and J. Gauvain, (2006), Multistage speaker diarization of broadcast news. IEEE Transactions on Audio, Speech, and Language Processing, VOL. 14, NO.5, SEPTEMBER 2006.
54. D. Reynolds and P. Torres-Carrasquillo, Approaches and applications of audio diarization, in Proc. Int. Conf. Acoust., Speech, Signal Process, Philadelphia, PA, 2005, pp. 953–956.