Odyssey 2016
21-24 Jun 2016, Bilbao, Spain

On the Use of PLDA i-vector Scoring for Clustering Short Segments

Itay Salmun[1,2], Irit Opher[1], Itshak Lapidot[1]

[1]Afeka Center for Language Processing (ACLP)
Afeka Tel-Aviv Academic College of Engineering, Israel
[2]Department of Electrical and Computer Engineering
Ben-Gurion University of the Negev, Beer-Sheva, Israel
itaysa@afeka.ac.il, irito@afeka.ac.il, itshakl@afeka.ac.il

Abstract

This paper extends upon a previous work using Mean Shift algorithm to perform speaker clustering on i-vectors generated from short speech segments. In this paper we examine the effectiveness of *probabilistic linear discriminant analysis* (PLDA) scoring as the metric of the mean shift clustering algorithm in the presence of different number of speakers. Our proposed method, combined with *k-nearest neighbors* (kNN) for bandwidth estimation, yields better and more robust results in comparison to the cosine similarity with fixed neighborhood bandwidth for clustering segments of large number of speakers. In the case of 30 speakers, we achieved evaluation parameter K of 72.1 with the PLDA-based mean shift algorithm compared to 65.9 with the cosine-based baseline system.

Index Terms: Speaker clustering, mean shift clustering, Probabilistic Linear Discriminant Analysis, two-covariance model, k-Nearest Neighbors, i-vectors, short segments.

1. Introduction

Speaker clustering is the task of identifying all segments from the same speaker in a set of speech segments. It is an inherent part of many speaker diarization algorithms [1] and it is also widely used for speaker adaptation [2]. In speaker diarization the conversation first undergoes a speaker change detection phase and then the speech segments are clustered into homogenous clusters, where each cluster should consist of a segments belonging to a single speaker. The common assumption is that during the conversation the channel/environment of each speaker is constant, which may facilitate the clustering process. In our case we address a different aspect of the clustering problem. The segments are well defined by a *push to talk* (ptt) button, however, the speech segments of the same speaker may arrive from different environments. One such example is a Taxi station speech recording system. Taxi drivers use a radio system for communication and usually the speech segments are very short. All the drivers use the same frequency and the same channel, so each recording of the communication includes multiple speakers on the same channel from multiple transmitting devices. Furthermore, each driver may drive different cars, and one car can be used by several drivers. As the speech is collected during several hours, the acoustic environment of each speaker can also change over this time period. Such high variability makes the speaker clustering task extremely challenging.

In [3], mean shift clustering algorithm based on cosine similarity was applied to i-vectors as a solution for generating single speaker clusters. This method was tested on the NIST 2008 Speaker Recognition database with segments of 2.5 seconds on average, and showed that a non-flat cost function is more appropriate than a flat one. Moreover, it was shown that the best results are obtained with i-vectors that were extracted with a 2048-mixture *universal background model* (UBM) and a *total variability* (TV) matrix of rank 400. In addition, it was shown that the *Within Class Covariance Normalization* (WCCN) does not improve the performance in a case of a UBM of size 2048. Therefore, all experiments in this paper were carried out with a 2048 UBM and a 400 TV. Moreover, no WCCN was performed.

Recently, i-vector extraction [2], [6] followed by Probabilistic Linear Discriminant Analysis (PLDA) [7], [8] has proven to yield state-of-the-art speaker verification performance. The i-vector is a low-dimensional fixed length vector extracted from all cepstral feature vectors that represent speech segment. By comparing two i-vectors corresponding to two segments a speaker verification score can be produced. This score is typically designed to give a good estimation of the log-likelihood ratio between the *same-speaker* and *different speaker* hypotheses. Good performance was reported when scores were computed as cosine similarity, better performance, nonetheless, is obtained with PLDA.

PLDA is similar to the Joint Factor Analysis (JFA) [9] approach but applied in the low-dimensional total variability space rather than the GMM supervector space [10]. Its main characteristic is that we split the total data variability into *within-individual* and *between-individual* variabilities, both residing on small-dimensional subspaces. This generative method has proven to be successful in the task of face recognition with uncontrolled conditions including variabilities in pose, lighting, and facial expressions [7]. Speaker clustering with variations in speaking style and acoustic environments is a similar problem. Hence, we choose the PLDA scoring as the new similarity measure between two i-vectors for the mean shift algorithm.

The paper is organized as follows: Section 2 recalls the mean shift algorithm; The PLDA approach, the two-covariance model and the integration with the mean shift algorithm are described in Section 3; Section 4 presents the clustering system; The experimental setup is discussed in Section 5; We present the experimental results in Section 6 and conclude the paper in Section 7.

doi: 10.21437/Odyssey.2016-59

2. The Mean Shift algorithm

The mean shift Algorithm is a non-parametric iterative algorithm that estimates the probability density function of a random variable [4]. Inspired by the Parzen window approach to non-parametric density estimation, the algorithm does not require prior knowledge of the number of clusters, and does not assume anything regarding the shape of the clusters. Dense regions in feature space correspond to local maxima or modes. So for each data point, we perform gradient ascent on the estimated local density until convergence is reached. The stationary points obtained via gradient ascent represent the modes of the density function. All points associated with the same stationary point belong to the same cluster.

2.1. Classic mean shift algorithm

The gradient of the density function $f(\phi)$ is required in order to find these modes. From the mathematical development in [4], [5], [11] and [12], the mean shift vector $m_h(\phi)$ expression is derived to be:

$$m_h(\phi) = \frac{\sum_{i=1}^{n} \phi_i g\left(\left\|\frac{\phi - \phi_i}{h}\right\|^2\right)}{\sum_{i=1}^{n} g\left(\left\|\frac{\phi - \phi_i}{h}\right\|^2\right)} - \phi \tag{1}$$

where ϕ is the current position of the mean shift. When applying this algorithm to clustering of i-vectors, at the mean shift algorithm first iteration, ϕ is a randomly selected i-vector from the pool of i-vectors. ϕ_i is the i-vectors to be clustered, h is the bandwidth and $g(\phi)$ is the kernel. The mean shift vector $m_h(\phi)$ is the difference of the current position (for instance vector ϕ) and the next position presented by the weighed sample mean vector of the neighborhood. The weights in the mean shift vector formula are given by the binary outputs (i.e. 0 or 1) of the flat kernel $g(\phi)$. For simplicity, we denote the uniform kernel with bandwidth h by $g(\phi,\phi_i,h)$ so that:

$$g(\phi,\phi_i,h) = \begin{cases} 1 & \|\phi - \phi_i\|^2 \le h^2 \\ 0 & \|\phi - \phi_i\|^2 > h^2 \end{cases} \tag{2}$$

i.e. $g(\phi,\phi_i,h)$ selects a subset $S_h(\phi)$ of n_ϕ i-vectors in which the Euclidean pairwise distances with ϕ are less or equal to the threshold (bandwidth) h:

$$S_h(\phi) \equiv \{\phi_i : \|\phi - \phi_i\| \le h\} \tag{3}$$

the iterative processing of calculating the sample mean followed by data shifting converges to a mode of the data distribution.

2.2. Improvements to the classic mean shift algorithm

The classic mean shift algorithm based on a flat kernel relies on the Euclidean distance for finding points falling within the window as shown in eq. (3). In a previous work, the Cosine metric was proposed instead of the Euclidean one to construct a new version of the mean shift algorithm [13]. This approach was found to be more appropriate for speaker clustering. The cosine similarity between two vectors is given by:

$$D(\phi_i,\phi_j) \equiv \left(\frac{\phi_i \cdot \phi_j}{\|\phi_i\|\|\phi_j\|}\right) = \cos(\theta) \tag{4}$$

in order to use the cosine similarity in the mean shift algorithm, only one modification has been introduced:

$$S_h(\phi) \equiv \{\phi_i : D(\phi,\phi_i) \ge h\} \tag{5}$$

where $D(\phi,\phi_i)$ is the cosine similarity between ϕ and ϕ_i given by the formula. This corresponds to redefining the uniform kernel as:

$$g(\phi,\phi_i,h) = \begin{cases} 1 & D(\phi,\phi_i) \ge h \\ 0 & D(\phi,\phi_i) < h \end{cases} \tag{6}$$

This modification was tested in [3] where it was found that using a non-uniform cosine similarity kernel yields better results than using a flat kernel. This kernel used in [3] was:

$$g(\phi,\phi_i,h) = \begin{cases} D(\phi,\phi_i)^4 & D(\phi,\phi_i) \ge h \\ 0 & D(\phi,\phi_i) < h \end{cases} \tag{7}$$

For the cosine similarity, there exist lower and upper bounds of the distance parameter, so it is easy to tune the bandwidth parameter h as it bound to be in the range of ± 1. For other types of similarities or scores, like the PLDA score, the range is not necessary known. For that reason, adaptive mean shift is used where the bandwidth parameter vary for each data point (i-vector) [14]. The h parameter is calculated using k-Nearest Neighbor (kNN) algorithm. If ϕ_{ik} is the k nearest neighbor of ϕ_i then the bandwidth is calculated as:

$$h_i = \|\phi_i - \phi_{ik}\| \tag{8}$$

while instead of Euclidian distance, the two-covariance scoring is used:

$$h_i = s(\phi_i,\phi_{ik}) \tag{9}$$

this scoring will be discussed in detail in a subsequent section.

3. PLDA

Probabilistic Linear Discriminant Analysis (PLDA) [7] is one of the most successful models for i-vectors comparison. In PLDA, speaker and session variability is modeled with separate subspaces in order to tease apart the contribution of the session variability from that of the speaker's. The fixed length nature of i-vectors allows this to be done in a relatively easier manner than in the acoustic space. In this section we briefly recall the PLDA framework and the *two-covariance model* that will be used in the mean shift clustering algorithm.

3.1. Standard PLDA

The i-vector generation process can be described by means of a latent variable probabilistic model where i-vector ϕ is modeled as the sum of three factors, namely a speaker factor y, an inter-session (channel) factor x and the residual noise ε as:

$$\phi = \mu + Vy + Ux + \varepsilon \quad (10)$$

Here, $\mu + Vy$ is the speaker-dependent part, and $Ux + \varepsilon$ is the channel-dependent part. V is a set of basis vectors for the speaker subspace, representing *between-speaker* variability, and U is a set of basis vectors for the channel subspace, representing *within-speaker* variability. The generation of an i-vector requires choosing a random speaker factor y according to the speaker prior distribution $p(y)$ and a random inter-session factor x according to a prior distribution $p(x)$. The i-vector is the sum of $Vy + Ux$, the mean vector μ and of the residual noise ε generated according to the distribution $p(\varepsilon)$.

PLDA estimates the matrices V, U, which maximize the likelihood of the observed i-vectors, assuming that i-vectors from the same speaker share the same speaker factor, i.e. the same value for latent variable y.

The simplest PLDA model assumes a Gaussian distribution for the prior parameters (G-PLDA). Nonetheless, in [8] it is shown that Maximum Likelihood estimation of the PLDA parameters under a Gaussian assumption fails to produce accurate models for i-vectors. Thus, Heavy-Tailed distributions for the model priors have been proposed leading to the Heavy-Tailed PLDA model (HT-PLDA) that resulted in improved performance, which however, is computationally expensive.

A simpler non-linear transformation of the i-vectors was proposed in [15]. This approach preserves the Gaussian distribution assumption, but incorporates a pre-processing step which involves whitening the i-vectors followed by normalizing their length. This technique, called *radial Gaussianisation*, restores the Gaussian assumptions of the PLDA model. Using these normalized i-vectors, the performance of the Heavy-Tailed and Gaussian PLDA models is comparable, the latter being much faster both in training and testing [16].

3.2. Two-covariance model

In this work we used more simplified model. This model is obtained by merging together the residual noise and their inter-session components, assuming that the speaker and inter-session subspaces span the entire i-vector subspace. This simplified model is referred to as the *two-covariance model* [17]. The two-covariance model is a generative linear-Gaussian model, where latent vectors y representing speakers are assumed to be distributed according to prior distribution:

$$p(y) = N\left(y; \mu, B^{-1}\right) \quad (11)$$

where B^{-1} is the between-speaker covariance matrix, and the distribution of the i-vector given the speaker identity is also Gaussian:

$$p\left(\phi | y\right) = N\left(\phi; y, W^{-1}\right) \quad (12)$$

where W^{-1} is the within-speaker covariance matrix. Moreover, both matrices B and W are *full* precision matrices. Thus, unlike the standard PLDA model, we no longer have any subspaces with reduced dimensionality. The Maximum Likelihood estimates of the model parameters μ, B^{-1} and W^{-1} can be obtained using EM algorithm as in [17]. Given a set of n i-vectors associated to the same speaker, the posterior of y is also Gaussian:

$$p(y|\phi_1,...,\phi_n) = N\left(y \mid \mathrm{L}^{-1}\gamma, \mathrm{L}^{-1}\right) \quad (13)$$

and the parameters of the distribution are:

$$\begin{aligned} L &= B + nW \\ \gamma &= B\mu + W\sum\phi \end{aligned} \quad (14)$$

3.3. Two-covariance scoring

We are following the analysis of two-covariance scoring as in [18]. The conditional likelihood of the i-vectors ϕ_1 and ϕ_2 allows obtaining the speaker verification log-likelihood ratio score between the *same-speaker* hypothesis H_s and *different-speaker* hypothesis H_d:

$$\lambda = \log\frac{p(\phi_1,\phi_2 \mid H_s)}{p(\phi_1,\phi_2 \mid H_d)} \quad (15)$$

where the numerator probability is computed assuming that the i-vectors ϕ_1 and ϕ_2 belong to the same speaker, i.e. they share a common value of the hidden variable y. According to Bayes rule this probability can be computed as:

$$p(\phi_1,\phi_2 \mid H_s) = \frac{p(\phi_1,\phi_2 \mid y_0)\mathrm{p}(y_0)}{p(y_0 \mid \phi_1,\phi_2)} \quad (16)$$

and y_0 is any value which does not cause the denominator to be zero. Since the intersession variability components of different segments are assumed to be conditionally independent, i.e. the i-vectors are independent given the speaker variable, and can be rewritten as:

$$p(\phi_1,\phi_2 \mid H_s) = \frac{p(\phi_1 \mid y_0)p(\phi_2 \mid y_0)\mathrm{p}(y_0)}{p(y_0 \mid \phi_1,\phi_2)} \quad (17)$$

where the denominator of (17) is computed as the i-vectors ϕ_1 and ϕ_2 belong to different speakers, as:

$$\begin{aligned} p(\phi_1,\phi_2 \mid H_d) &= p(\phi_1)\cdot p(\phi_2) = \\ &= \frac{p(\phi_1 \mid \mathrm{y}_0)p(\mathrm{y}_0)}{p(\mathrm{y}_0 \mid \phi_1)} \cdot \frac{p(\phi_2 \mid \mathrm{y}_0)p(\mathrm{y}_0)}{p(\mathrm{y}_0 \mid \phi_2)} \end{aligned} \quad (18)$$

the first equality is derived from the independence of speaker factors, where the second one is derived from Bayes rule. After the extensions of the numerator and denominator we get:

$$\lambda = \log\frac{p(y_0 \mid \phi_1)p(y_0 \mid \phi_2)}{p(y_0)p(y_0 \mid \phi_1,\phi_2)} \quad (19)$$

using (11) and (13) and selecting $y_0 = 0$, we finally get the log-likelihood ratio:

$$\begin{aligned} \lambda = \frac{1}{2}(&\log\left|\tilde{\Gamma}\right| - \gamma_1^T\tilde{\Gamma}\gamma_1 + \log\left|\tilde{\Gamma}\right| - \gamma_2^T\tilde{\Gamma}\gamma_2 - \log|B| \\ &- \mu^T\tilde{\Gamma}\mu - \log\left|\tilde{\Lambda}\right| + \gamma_{1,2}^T\tilde{\Lambda}\gamma_{1,2}) \end{aligned} \quad (20)$$

where according to (14):

$$\begin{aligned}\tilde{\Lambda} &= (B+2W)^{-1}\\ \tilde{\Gamma} &= (B+W)^{-1}\\ \gamma_{1,2} &= B\mu + W(\phi_1+\phi_2)\\ \gamma_i &= B\mu + W\phi_i\end{aligned} \tag{21}$$

if we gathered all the terms that are not a function of γ_1, γ_2 and $\gamma_{1,2}$ in a constant $\tilde{k}$ we receive:

$$\lambda = \frac{1}{2}\left(\tilde{k} + \gamma_{1,2}^T\tilde{\Lambda}\gamma_{1,2} - \gamma_1^T\tilde{\Gamma}\gamma_1 - \gamma_2^T\tilde{\Gamma}\gamma_2\right) \tag{22}$$

with

$$\tilde{k} = 2\log\left|\tilde{\Gamma}\right| - \log\left|\tilde{B}\right| - \log\left|\tilde{\Lambda}\right| + \mu^T B\mu \tag{23}$$

By substituting (21) in (22) we obtain the score between two i-vectors:

$$\begin{aligned}s(\phi_1,\phi_2) = \frac{1}{2}\big(&(B\mu + W(\phi_1+\phi_2))^T\tilde{\Lambda}(B\mu + W(\phi_1+\phi_2))\\ &-(\mathrm{B}\mu + \mathrm{W}\phi_1)^T\tilde{\Gamma}(\mathrm{B}\mu + \mathrm{W}\phi_1)\\ &-(\mathrm{B}\mu + \mathrm{W}\phi_2)^T\tilde{\Gamma}(\mathrm{B}\mu + \mathrm{W}\phi_2) + \tilde{k}\big)\end{aligned} \tag{24}$$

which can be rewritten as:

$$\begin{aligned}s(\phi_1,\phi_2) = {}&\phi_1^T\tilde{\Lambda}\phi_2 + \phi_2^T\tilde{\Lambda}\phi_1 + \phi_1^T\tilde{\Gamma}\phi_1 + \phi_2^T\tilde{\Gamma}\phi_2\\ &+(\phi_1+\phi_2)^T c + k\end{aligned} \tag{25}$$

the relation to the original model parameters are according to:

$$\begin{aligned}\Lambda &= \frac{1}{2}W^T\tilde{\Lambda}W\\ \Gamma &= \frac{1}{2}W^T(\tilde{\Lambda}-\tilde{\Gamma})W\\ c &= W^T(\tilde{\Lambda}-\tilde{\Gamma})B\mu\\ k &= \tilde{k} + \frac{1}{2}\left((B\mu)^T(\tilde{\Lambda}-2\tilde{\Gamma})B\mu\right)\end{aligned} \tag{26}$$

3.4. PLDA score in mean shift algorithm

In the clustering process, the mean shift algorithm shifts a certain i-vector to a point that is more likely to be a local maxima of the estimated density. This step is repeated until convergence. In this work, the two-covariance model is used for computing the score between any two i-vectors. High scores mean that it is more likely that the two i-vectors share the same speaker factor, i.e. the same value for latent variable y. In our implementation, the update of each mean shift iteration, for each data point, will be according to the PLDA score between the specific data point and all other data points. The PLDA two-covariance score between two i-vectors $s(\phi,\phi_i)$ replaces the cosine similarity in (4) with the equation in (25), so (5) turns to be:

$$S_{h_i}(\phi) \equiv \{\phi_i : s(\phi,\phi_i) \geq h_i\} \tag{27}$$

In other words, we select a subset $S_h(\phi)$ of data points in which the PLDA pairwise score with ϕ are larger or equal to the adaptive bandwidth h_i as explained in (9). With mean shift weighted kernel of:

$$g(\phi,\phi_i,h_i) = \begin{cases} s(\phi,\phi_i) & s(\phi,\phi_i) \geq h_i \\ 0 & s(\phi,\phi_i) < h_i \end{cases} \tag{28}$$

4. The clustering system

Before performing clustering, train in advance the UBM and TV matrix for i-vectors extraction. Then, train the PCA matrix T, and the whitening transformation matrix C. The low rank i-vectors are:

$$\varphi = \frac{C\mathrm{T}\phi}{\left\|C\mathrm{T}\phi\right\|} \tag{29}$$

Using the low rank normalized i-vectors, train the two-covariance model parameters, i.e. W^{-1} and B^{-1}. Then, given a set of speech segments, cluster them according to the following steps:

1. Extract the i-vectors for all the speech segments $\{\phi_i\}$
2. Find all modes $\{m_i\}$ of the data with the two-covariance based mean shift algorithm, using (25) on low rank normalized i-vectors $s(\varphi_1,\varphi_2)$
3. Merge all shifted points, that represent the modes, according to Euclidian distance with fixed threshold: $\|m-m_i\| \leq Th$

5. Experimental setup

5.1. Test data

The experiments were carried out on the NIST 2008 Speaker Recognition Database [19], [20] and [21]. The test corpus *short2-short3-Test7* was used to extract only male speakers for clustering. This test includes 5 minutes English telephone speech segments that were cut into small speech segments of about 2.5 seconds each. On the average, about 34 short segments were extracted per speaker. Figure 1 shows the distribution of segments' length and the distribution of the number of segments per speaker that were generated.

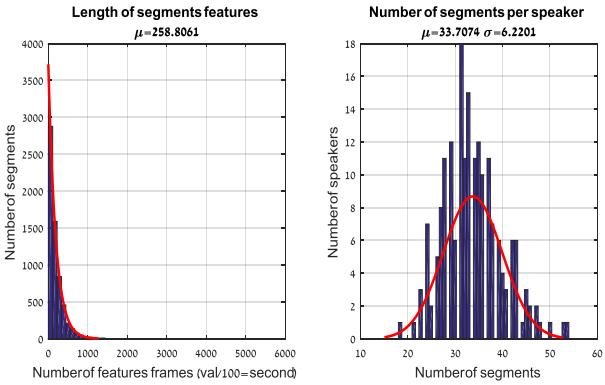

Figure 1: *segments length distribution (on the left), and the distribution of the number of segments per speaker (on the right)*

5.2. Feature extraction and training data

In our experiments we used *Mel frequency cepstral coefficients* (MFCC), that were extracted using a 25ms Hamming window. The MFCCs were calculated using a 19-channel mel-frequency filterbank together with log energy, and calculated every 10ms. Delta and delta delta coefficients were then calculated using a 5 frame window to produce 60-dimensional feature vectors. Mean subtraction and variance normalization were applied due to the assumption that speech segments can be recorded at different environments.

We used a male only UBM containing 2048 Gaussians. This UBM was trained with the LDC releases of Fisher Part 1; Switchboard II, Phase 2; switchboard Cellular, Parts 1 and 2; and NIST 2004-2006 SRE. The male only total variability matrix with 400 dimensions was trained on labeled data from same databases as for the UBM. In total, we used 975 unique male speakers with 10705 sessions.

PLDA model, PCA eigenvectors matrix and whitening transformation matrix were trained with the same data as for the total variability matrix. We found that for short segments clustering, training the PLDA matrices on long utterances leads to results degradation. Hence, we split the utterances into short segments of about the same duration as the segments to be clustered. For the PLDA scoring, we found that whitening the i-vectors followed by dimension reduction from 400 to 250 using PCA, improves the clustering results.

5.3. Baseline system

Our baseline system is designed according to [3]. The mean shift clustering algorithm is based on the cosine similarity with non-uniform cosine kernel as in (7), and has fixed threshold for the bandwidth parameter h. Moreover, it uses random selecting point configuration which randomly selects a point for shifting and skips all other points in the neighborhood [11], which will be called from now on "Random mean shift".

5.4. Number of speakers

In all the experiments, we fixed the number of speakers to be 30. Only in section 6.6 we validate the robustness of the proposed approach, i.e. PLDA based mean shift, by clustering different number of speakers.

6. Results and Discussions

In this section we present the evaluation criteria and the experiments which were conducted.

6.1. Evaluation criterion

We use the *purity* concept explained in [22] to calculate both the *average cluster purity* (ACP) and *average speaker purity* (ASP). ASP is a measure for how well a speaker is limited to only one cluster, while ACP is a measure for how well a cluster is limited to only one speaker. For ease comparison between systems, the geometrical mean of ASP and ACP is used to obtain an overall evaluation criterion:

$$K = \sqrt{ACP * \mathrm{ASP}} \tag{30}$$

Moreover, we use the *average number of detected speakers* (ANDS) criterion as extra information of the clustering performance.

6.2. Controlling the bandwidth parameter h

The bandwidth parameter h may significantly affect the clustering results. Narrow kernel bandwidth will result in a large number of clusters, i.e. speakers. A wide bandwidth yields exactly the opposite. A comparison between using a fixed threshold for the bandwidth and an adaptive threshold as in (9) is presented in Fig. 2.

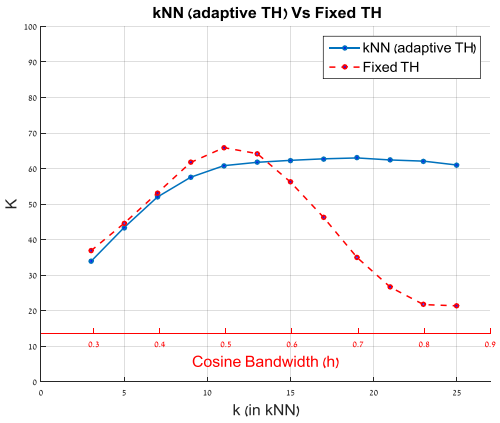

Figure 2: *comparing performance of cosine based random mean shift clustering: adaptive threshold using kNN Vs a fixed threshold*

Although using a fixed threshold for the bandwidth may sometimes outperform the adaptive threshold, it does outperform it in a narrow bandwidth range. This sensitive behavior of the fixed bandwidth parameter causes its tuning to be much harder, resulting in poor clustering performance. Hence, adaptive threshold based kNN is used.

6.3. Mean Shift's selecting point configuration

As we have seen that using an adaptive threshold based kNN is more robust to changes, from now on, only the kNN based bandwidth will be used. Two different types of mean shift configurations are under examination in the following experiment; *random* mean shift with random selecting point as in the baseline system, and *full* mean shift where all points are selected for shifting [11].

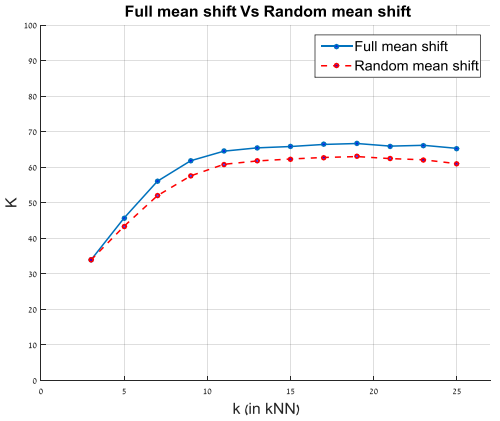

Figure 3: *comparing performance of cosine based mean shift clustering with adaptive threshold: full mean shift Vs random mean shift*

It can be seen that for any k given, the full mean shift clustering outperforms the random mean shift clustering.

6.4. PLDA based mean shift

As the full version of the mean shift shows better clustering performance than the random version, from now on, the full mean shift will be applied. The PLDA two-covariance model is used for computing the score between each pair of i-vectors as demonstrated in (25). According to the PLDA scores, the bandwidth is set as in equation (9) and the current mean is shifted. This is done on mean shift configuration with adaptive threshold (kNN).

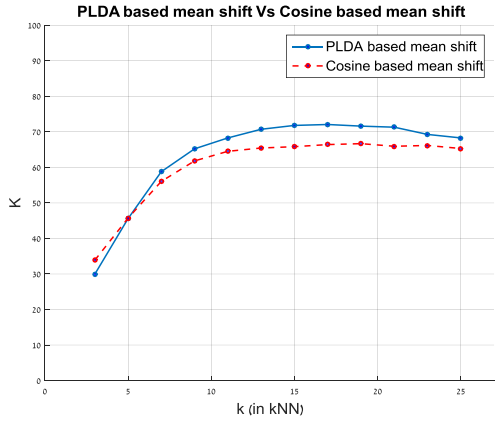

Figure 4: *comparing performance of full mean shift clustering with adaptive threshold: PLDA based mean shift Vs cosine based mean shift*

When PLDA scoring defines the similarity for the mean shift algorithm, it provides better K value than the system with cosine kernel.

6.5. PLDA training

As mentioned in section 5.2, we found that for short segments clustering, training the PLDA matrices on long utterances of 5 minutes leads to results degradation. Fig. 5 shows the comparison between the PLDA mean shift with short segments versus long segments training.

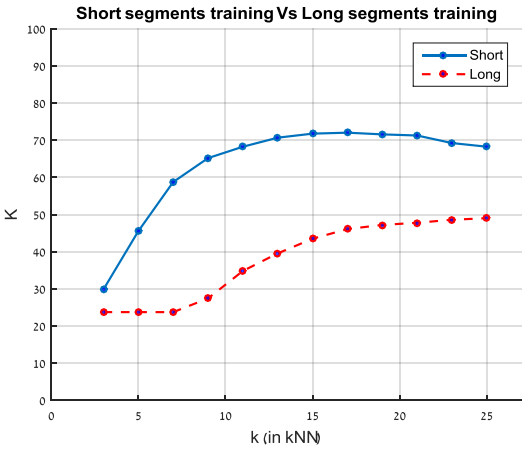

Figure 5: *comparing performance of PLDA based mean shift: PLDA model trained on short segments Vs PLDA model trained on long segments*

The degradation in the case of the lengths mismatch can be explained by the fact that the conditions of the PLDA training greatly differ from the clustering conditions (5 minutes per utterance versus 2.5 seconds). The short segments length of 2.5 seconds is not sufficient for a good i-vector extraction, so it differs from the 5 minutes-based i-vector; The *between-speaker* subspace and *within-speaker* subspace are different in that case too, therefore the two-covariance trained model does not provide a good approximation for the log-likelihood ratio between the *same-speaker* and *different speaker* hypotheses on short segments clustering.

The difference can also be seen by the distribution of the PLDA scores. When trained on longer segments, PLDA model yielded significantly lower scores. The high variance of the scores implies that the mean shift algorithm has difficulties to converge.

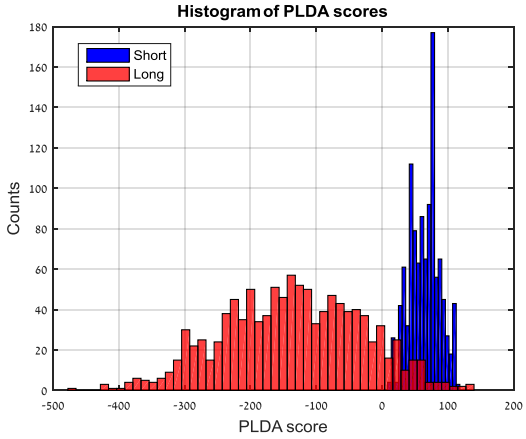

Figure 6: *Histogram of PLDA scores: PLDA model trained on short segments Vs PLDA model trained on long segments*

6.6. Different number of speakers

The proposed approach for testing the robustness of our method is carried out according to the experiments described in sections 6.2 – 6.4 and includes PLDA based mean shift with adaptive bandwidth estimation and full point selection.

We tested the robustness of the proposed approach for clustering different number of speakers. For a large number of speakers, the density of the i-vectors becomes high; this is also true for the modes which become closer to one another. Due to this fact, it is more difficult to distinguish between the speakers.

For each experiment, we fixed the bandwidth parameter h or k that maximizes the evaluation criterion K (the baseline system uses fixed bandwidth while the proposed system uses kNN). Table 1 summarizes the results for the baseline system with cosine based mean shift while Table 2 summarizes it for the proposed PLDA based mean shift.

It can be seen that PLDA based mean shift is relatively robust to the number of speakers in the test. The k parameter of kNN can be fixed to 17 and it will perform well for any number of speakers, while in the baseline system the dependence on the tuning of the bandwidth parameter is much more crucial. Moreover, it will result in substantially more accurate estimation of the number of speakers in comparison to the cosine based mean shift. For the PLDA based mean shift, the ANDS is approximately 50% higher than the true

number of speakers in the experiment, in comparison to the baseline system where the ANDS is significantly higher. For example, given a test with 60 speakers, the PLDA clustering results in reasonable 90 clusters, where the baseline system results in 614 clusters. It can explain why ACP is usually better in the baseline system. It is due to the fact that there are many clusters which contain one segment only.

Table 1: *Results for different number of speakers for the cosine based mean shift* ***(baseline system)***

Number of Speakers	h	ACP	ASP	K	ANDS
3	0.35	92.2	80.1	85.7	6.1
7	0.40	89.5	71.6	79.9	21.1
15	0.45	77.6	63.3	70.0	60.6
22	0.50	85.0	57.6	69.9	136.6
30	0.50	81.7	53.2	65.9	195.0
60	0.55	84.6	44.3	61.2	614.1
188	0.55	68.4	42.8	54.1	1742.1

Table 2: *Results for different number of speakers for the PLDA based mean shift* ***(proposed system)***

Number of Speakers	k	ACP	ASP	K	ANDS
3	19	90.0	71.3	79.8	5.0
7	17	84.8	67.5	75.5	11.2
15	15	86.6	63.6	74.1	26.9
22	15	86.6	65.3	75.1	36.4
30	17	80.8	64.3	72.1	46.6
60	17	73.8	61.1	67.2	90.0
188	17	61.4	53.1	57.1	283.0

6.7. Comparison of mean shift configurations

In this section we summarize the different system configurations which were under examination. The performance is presented in Fig. 7, where all results correspond to 30 speakers.

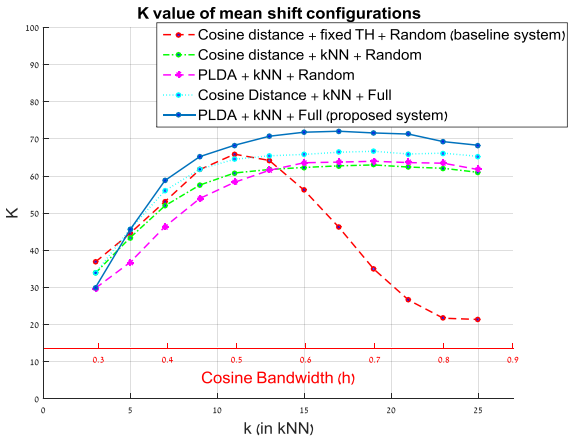

Figure 7: *comparing K value of mean shift configurations*

It is evident that the proposed system performs better than all other configurations. While the kNN adaptive threshold improves the robustness of the system, the PLDA based mean shift yields higher K values than the cosine. It can be seen that the baseline system with cosine kernel can provide comparable results but in a narrow range. This lack of robustness can be seen also in the average number of detected speakers as shown in Fig. 8.

The proposed PLDA based system shows almost no degradation in K and ANDS values as the k of kNN moves away from the optimal point. This behavior can be seen by the flat curve in the range of $k=[11,25]$, where in that range $K \cong 72$ and $ANDS \cong 50$.

According to the baseline system results, $h=0.50$ provides the highest K but at that point the number of detected speakers is 195. In order to get the same number of speakers as in the proposed system, we have to choose $h=0.40$. For this value of the bandwidth, the clustering process results in about 50 detected speakers but with a very large degradation to $K=53$, in compassion to $K=72$ in the proposed one.

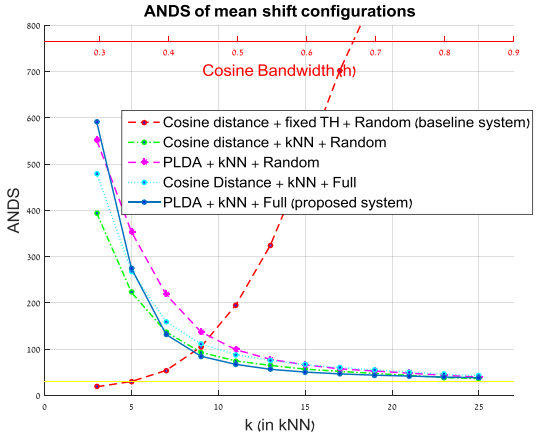

Figure 8: *comparing the average number of detected speakers (ANDS) of mean shift configurations. The horizontal line at the bottom displays the real number of speakers (30)*

7. Conclusions

In this paper, we extended a previous work addressing the problem of short segments speaker clustering. We introduced the use of PLDA two-covariance scoring as the similarity measure for the Mean Shift algorithm and tested it on different number of speakers. We then compared its clustering performance to the cosine-based baseline system. Our analysis shows that it is better to use PLDA scoring over the same system with cosine scoring. Performance was further improved with a kNN based adaptive bandwidth and Mean Shift with Full point selection. It was shown that when PLDA models were trained on long utterances, compared with the length of the clustered segments, the system performs poorly.

While the proposed system is more time consuming, it outperforms the baseline system in the following aspects: it yields better results when clustering large numbers of speakers; it is more robust to changes in the number of speakers; no bandwidth adjustment is needed; and the average number of detected speakers is by far more accurate.

8. References

[1] S. Tranter and D. Reynolds, "An overview of automatic speaker diarization systems," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 14, no. 5, pp. 1557–1565, September 2006.

[2] N. Dehak, P. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, "Frontend factor analysis for speaker verification," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 4, May 2011, pp. 788-798.

[3] I. Shapiro, N. Rabin, I. Opher, and I. Lapidot, "Clustering short push-to-talk segments,"Interspeech'15, September 6-10, 2015, Dresden, Germany.

[4] K. Fukunaga and L. Hostetler, "The estimation of the gradient of a density function, with applications in pattern recognition," *IEEE Transactions. on Information Theory*, vol. 21, no. 1, pp. 32–40, January 1975.

[5] D. Comaniciu and P. Meer, "Mean shift: A robust approach toward feature space analysis," *IEEE Transaction Pattern Analysis and Machine Intelligence*, vol. 24, no. 5, pp. 603-619, May 2002.

[6] P. Kenny, "Joint factor analysis of speaker and session variability: theory and algorithms". Technical report CRIM-06/08-14, January 2006.

[7] Simon J.D. Prince and James H. Elder, "Probabilistic linear discriminant analysis for inferences about identity," in International Conference on Computer Vision. IEEE,2007, pp. 1–8.

[8] P. Kenny, "Bayesian speaker verification with heavy–tailed priors," in Keynote presentation, Odyssey 2010, The Speaker and Language. Recognition Workshop, 2010.

[9] P. Kenny, G. Boulianne, P. Ouellet, and P. Dumouchel, "Speaker and session variability in GMM-based speaker verification," *IEEE Transaction Audio, Speech, Language Processing*, vol. 15, no. 4, pp. 1448–1460, May 2007.

[10] Reynolds, D., Quatieri, T., Dunn, R., 2000. Speaker verification using adapted Gaussian mixture models. Digital Signal Process. 10 (1–3), 19–41.

[11] M. Senoussaoui, P. Kenny, T. Stafylakis, and P. Dumouchel. "A study of the cosine distance-based mean shift for telephone speech diarization," *Audio, Speech, and Language Processing, IEEE/ACM Transactions on*, vol. 22, no.1 pp., 217-227, 2014.

[12] D. Comaniciu, V. Ramesh, and P. Meer, "Kernel-based object tracking". *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no.5 pp., 564–577, May 2003.

[13] M. Senoussaoui, P. Kenny, P. Dumouchel, and T. Stafylakis, "Efficient Iterative Mean Shift based Cosine Dissimilarity for Multi-Recording Speaker Clustering," *in IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, May 26-31, 2013, Vancouver, Canada, 2013, pp. 4311-4315.

[14] D. Comaniciu, V. Ramesh, and P. Meer, "The variable bandwidth mean shift and data-driven scale selection," in Proceedings 8th International Conference on Computer Vision, Vancouver, Canada, volume I, July 2001, pp. 438–445.

[15] Garcia-Romero, D., Espy-Wilson, C., 2011. Analysis of i-vector length normalization in speaker recognition systems. In: Proceedings of Interspeech.pp. 249–252.

[16] P.-M. Bousquet, J.-F. Bonastre, and D. Matrouf, "Exploring some limits of gaussian plda modeling for i-vector distributions," in Odyssey: The Speaker and Language Recognition Workshop, 2014.

[17] A. Sizov, Aleksandr, Kong Aik Lee, and Tomi Kinnunen. "Unifying Probabilistic Linear Discriminant Analysis Variants in Biometric Authentication." *Structural, Syntactic, and Statistical Pattern Recognition: Joint IAPR International Workshop, S+ SSPR* 2014, Joensuu, Finland, August 20-22, 2014, Proceedings. Vol. 8621. Springer, 2014.

[18] S. Cumani, N. Brummer, L. Burget, P. Laface, O. Plchot, and V. Vasilakakis, "Pairwise discriminative speaker veri- fication in the i-vector space," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 21, no. 6, pp. 1217–1227, 2013.

[19] NIST Multimodal Information Group. 2008 NIST Speaker Recognition Evaluation Training Set Part 1 LDC2011S05. Philadelphia: Linguistic Data Consortium, 2011. Online at: https://catalog.ldc.upenn.edu/LDC2011S05

[20] NIST Multimodal Information Group. 2008 NIST Speaker Recognition Evaluation Training Set Part 2 LDC2011S07. Philadelphia: Linguistic Data Consortium, 2011. Online at: https://catalog.ldc.upenn.edu/LDC2011S07

[21] NIST Multimodal Information Group. 2008 NIST Speaker Recognition Evaluation Test Set LDC2011S08. Philadelphia: Linguistic Data Consortium, 2011. Online at: https://catalog.ldc.upenn.edu/LDC2011S08

[22] J. Ajmera, H. Bourlard, I. Lapidot and I. McCowan, "Unknown-multiple speaker clustering using HMM" *in Intl. Conf. on Spoken Language Proc*essing, 2002.