Odyssey 2014:
The Speaker and Language Recognition Workshop
16-19 June 2014, Joensuu, Finland

# A Latent Dirichlet Allocation Based Front-End for Speaker Verification

*Yusuf Ziya Işık[1,2], Hakan Erdogan[2], Ruhi Sarıkaya[3]*

[1]TUBITAK BILGEM , Gebze, Turkey
[2]Faculty of Engineering and Natural Sciences, University of Sabancı, Turkey
[3]Microsoft Corporation, Redmond, WA, USA

yusuf.ziya@tubitak.gov.tr, haerdogan@sabanciuniv.edu,ruhi.sarikaya@microsoft.com

## Abstract

Latent Dirichlet Allocation is a powerful topic model used heavily in natural language processing, image processing and biomedical signal processing fields to discover hidden structures behind observed data. In this work, we have adopted a variant of LDA for continuous descriptor vectors and use this model as a front-end for speaker verification similar to popular i-vector front-end. We have proposed an efficient hierarchical acoustic vocabulary creation method and presented a speaker verification system using latent topic probability features obtained using LDA front-end. We analysed the performance of the LDA front-end for various vocabulary and topic sizes, and obtained encouraging results on NIST SRE corpora. The proposed system is shown to improve the performance of an i-vector-PLDA baseline system when tested on NIST SRE12 corpora.

## 1. Introduction

Probabilistic topic models try to discover the hidden semantic structure in a collection of documents, and make it more convenient to explore and browse the documents in a collection. Using a probabilistic topic model we can easily find similar documents to a reference document in a collection, or categorize documents according to their semantic content. One of the most widely used topic models is Latent Dirichlet Allocation (LDA) [1, 2]. LDA is a hierarchical topic model that uses bag-of-words assumption for documents. A document in the collection can be about several topics, so each document has a distribution over the topics. The topics, themselves, are modeled with a distribution over words in the corpus.

LDA has been used for analyzing many types of text corpora such as newspapers, scientific journals, or tweets. In addition to text data, LDA has also been used for image and video data for tasks such as object recognition and novelty detection. To apply LDA to data with continuous valued observations, first a "visual vocabulary" is learned using methods such as k-means or kd-trees. Each continuous valued observation is then mapped to a "visual word" and the document (e.g., image) is transformed into a text document with discrete word observations. The standard LDA algorithm is then applied to the generated text corpora. Recently, approaches incorporating the generation of continuous descriptors into the LDA model have been proposed [3, 4]. These approaches apply a soft alignment of continuous descriptors to words in the vocabulary instead of hard assigning them.

LDA model has also been used for speech corpora. In [5], it is used for a language recognition task. First phone n-gram sequences are extracted from utterances and then LDA is applied by using n-gram symbols as words in a vocabulary. In [6], it is used for spoken document retrieval, and in [7] for computing spoken document similarity.

In all of the previous work, LDA is applied after a word or phone recognition step. In this paper, we use LDA for speaker verification for the first time. Instead of transcribing speech data in phone or word level prior to LDA modeling, we directly apply LDA to continuous valued local descriptors like MFCC vectors. We propose a technique to construct "acoustic" vocabularies which helps us to map the continuous descriptors to discrete acoustic words efficiently even for moderately large vocabulary sizes. Continuous descriptor vectors are soft aligned with acoustic words to obtain posterior counts. We will use LDA as a front-end to extract a fixed size topic distribution vector from each utterance. This use of LDA is similar to the use of popular i-vector model.

The rest of the paper is organized as follows: in section 2 we describe briefly the LDA model, and in section 3, we describe our acoustic vocabulary generation method. In section 4, we describe the overall speaker verification system using LDA generated topic features. In section 5, experimental results are given followed by the conclusions in the last section.

## 2. Latent Dirichlet Allocation Model

The generative process of the LDA for a corpus with $M$ documents and $K$ topics is given as below:

- For each of the $K$ topics, sample topic word distribution $\boldsymbol{\beta}_k \sim \text{Dirichlet}(\eta)$ where $\eta$ is the prior for Dirichlet distribution.
- For each document in the corpus, sample distribution over topics : $\boldsymbol{\theta}_m \sim \text{Dirichlet}(\alpha)$ where $\alpha$ is the prior for Dirichlet distribution.
- For each term in document $m$, sample topic index $z_{m,n} \sim \text{Multinomial}(\boldsymbol{\theta}_m)$ and using this sample term for word $w_{m,n} \sim \text{Multinomial}(\boldsymbol{\beta}_{z_{m,n}})$.

The graphical model of LDA can be seen in Fig. 1. Note that the number of topics $K$ is assumed to be known and held fixed. Usually symmetric priors are used, but in [8] it is found that using an asymmetric prior for document-topic distribution helps. The joint probability distribution for LDA is:

$$p(\mathcal{W}, \mathcal{Z}, \Theta, \boldsymbol{\beta}|\alpha, \eta) = \prod_{m=1}^{M} p(\boldsymbol{\theta}_m|\alpha) \prod_{n=1}^{N_m} p(w_{mn}|z_{mn}) p(z_{mn}|\boldsymbol{\theta}_m) \prod_{i=1}^{K} p(\boldsymbol{\beta}_i|\eta), \quad (1)$$

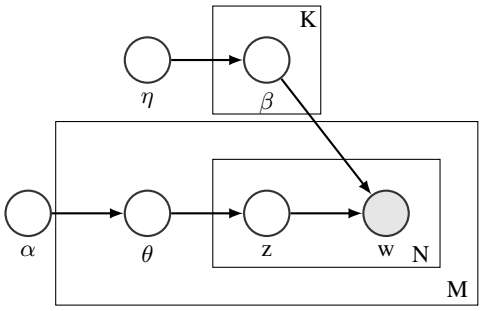


Figure 1: *Graphical model of the LDA model.*

where $\mathcal{W}$, $\mathcal{Z}$, and $\Theta$ are the words, topic indexes, and probability vector of documents over topics in the whole corpus, respectively. The main difficulty in LDA inference is that computing the posterior distribution of hidden variables is intractable. To solve this problem we can either use approximate inference methods such as mean field variational inference [1] or sampling techniques such as Gibbs sampling [9]. Using the variational alternative, we can approximate the posterior distribution of hidden variables as:

$$q(\mathcal{Z}, \Theta, \boldsymbol{\beta}) = \{\prod_{m=1}^{M}\prod_{n=1}^{N_m} q(z_{mn}|\boldsymbol{\phi}_{mn})\}\{\prod_{m=1}^{M} q(\boldsymbol{\theta}_m|\boldsymbol{\gamma}_m)\} \{\prod_{k=1}^{K} q(\boldsymbol{\beta}_k|\boldsymbol{\lambda}_k)\}. \quad (2)$$

where $q(z_{mn}|\boldsymbol{\phi}_{mn})$ is the posterior categorical distribution of the term $w_{mn}$ over the topics, $q(\boldsymbol{\theta}_m|\boldsymbol{\gamma}_m)$ is the posterior distribution of document $m$ over topics in the form of an asymmetric Dirichlet distribution with parameters $\boldsymbol{\gamma}_m$, and $q(\boldsymbol{\beta}_k|\boldsymbol{\lambda}_k)$ is the posterior distribution of the $k^{th}$ topic over terms in the form of an asymmetric Dirichlet distribution with parameters $\boldsymbol{\lambda}_k$. Note that even if we select a symmetric Dirichlet distribution for the priors, we use an asymmetric distribution for the posteriors. If we apply a variational EM algorithm to infer the posterior parameters, we obtain the following update equations:

$$\phi_{mwk} \propto \exp\left\{\psi(\gamma_{mk}) + \psi(\lambda_{kw}) - \psi(\sum_{j=1}^{V}\lambda_{kj})\right\} \quad (3)$$

$$\gamma_{mk} = \alpha + \sum_{w=1}^{V}\phi_{mwk} \quad (4)$$

$$\lambda_{kw} = \eta + \sum_{m=1}^{M} n_{mw}\phi_{mwk}, \quad (5)$$

where $n_{mw}$ is the count of term $w$ in document $m$, $V$ is the vocabulary size, and $\psi()$ is the di-gamma function. Note that for the multiple occurrences of the same word in a document, posterior distribution parameters are the same and calculated only once. Hence, we tie the distributions $\boldsymbol{\phi}_{mn}$ with the same term id $w$ in document $m$ into a single distribution $\boldsymbol{\phi}_{mw}$. This is an advantage of variational methods over sampling based inference methods.

For a new document not present in the training set, we can calculate the maximum a posteriori point estimate of document topic probability distribution vector $\theta$, by first iteratively applying equations 3 and 4 until convergence, and then using:

$$\hat{\theta}_k = \frac{\gamma_k}{\sum_{i=1}^{K}\gamma_i}. \quad (6)$$

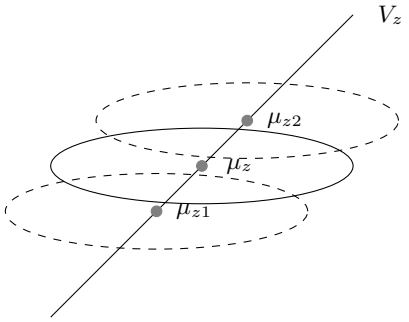


Figure 2: *Acoustic vocabulary generation process. Two child mixtures are generated from the $z^{th}$ UBM mixture with mean $\mu_z$. For each child mixture, the 1 dimensional i-vector $x$ is sampled from $\mathcal{N}(0,1)$ and the mean of the child mixture is obtained as $V_z x + \mu_z$. The child mixtures share the covariance matrix of their parent UBM mixture.*

This gives us the expected value of the posterior distribution $q(\theta|\gamma)$ and can be used as a fixed size feature vector for a given document.

## 3. An efficient acoustic vocabulary

Two important factors that may effect LDA performance is the size of the vocabulary $V$, and the number of topics $K$. Both are fixed prior to LDA modelling. In Section 5, we analyze the performance of our LDA based system for several values of $V$ and $K$. However, the value chosen for the vocabulary size has also a direct effect on the computation cost of our algorithm. A given descriptors' likelihood for every Gaussian in our acoustic vocabulary should be computed, and for large vocabulary sizes computational cost may be prohibitive. In image processing community, this is usually avoided using hierarchical vocabularies such as kd-trees and discretizing the continuous descriptor by hard assignment to only a single visual word. When using soft alignment of descriptors to acoustic words, we need an efficient procedure to compute the posterior counts of descriptors.

To efficiently produce acoustic vocabularies of various sizes and obtain posterior counts of acoustic words for a given descriptor vector, we will use the i-vector model and its corresponding universal background model (UBM). The i-vector model may be thought as a way to adapt a UBM to a given utterance. The generative story of the i-vector model is as follows:

- For each sequence, sample hidden factors $\mathbf{x}$ from $\mathcal{N}(\mathbf{x}; \mathbf{0}, \mathcal{I})$ once.
- For each observation in the sequence:
  - Sample mixture index $z$ from the multinomial distribution, $\mathrm{Mult}(\pi)$ where $\pi$ is the weight vector of the UBM,
  - Sample the observed variable $\mathbf{y}$ from $\mathcal{N}(\mathbf{y}; \mathbf{V}_z\mathbf{x} + \mu_z, \boldsymbol{\Sigma}_z)$ where $\mu_z$ and $\boldsymbol{\Sigma}_z$ are mean and covariance matrix of the $z^{th}$ mixture of UBM, and $\mathbf{V}_z$ is the factor loading matrix associated with that mixture.

Using the UBM and the i-vector model, we will build a two-level hierarchical acoustic vocabulary which will provide us an effective procedure to calculate posterior counts of words

in the vocabulary. At the top level of our hierarchical acoustic vocabulary, we have the UBM mixtures. For each UBM mixture, we generate child mixtures using the i-vector model. The number of child mixtures of a UBM mixture is proportional to its weight. Each child mixture shares the covariance matrix of its parent UBM mixture. When generating a child mixture, we follow the below steps:

- Randomly sample an i-vector $\mathbf{x}$ from $\mathcal{N}(\mathbf{x}; \mathbf{0}, \mathcal{I})$,
- Obtain the mean of the child mixture using: $\mathbf{V}_z\mathbf{x} + \mu_z$, where $z$ is the index of parent UBM mixture.

This procedure is demonstrated in Figure 2 for a single mixture of a UBM, where feature vectors are 2 dimensional and i-vectors are 1 dimensional. When calculating the posterior counts, we first apply top-N scoring using the UBM mixtures, and find the top performing $N$ mixtures of the UBM for each frame. Only the childs of these UBM mixtures are used for posterior count calculation for the given frame. Since i-vector model is a proven method to model the total variability space, we use it for generating acoustic words when large vocabularies are needed.

# 4. LDA front-end for speaker verification

Latent Dirichlet allocation model can be used as a feature extractor similar to the i-vector model. The i-vector model assumes that supervectors of Gaussian mixture models obtained by adapting a UBM lie in a low dimensional subspace. In [10], Kenny states that the coordinates of the representation learned by the i-vector model may be related to physical quantities constant for an utterance such as vocal tract length. Depending on the values of these physical quantities, some of the acoustic words may be active in an utterance (that is have high posterior counts), while others may be inactive. If we model co-occurences of the acoustic words in a corpus, we may obtain patterns related with these physical constants and their plausible combinations. Thus, LDA topic probability distributions may be a good candidate to preserve information regarding to the physical constants inherent in an utterance. For text documents the interpretation of topics is more intuitive. No work has been done in this study, to find and test interpretations of the topics learned by the LDA model. Instead, we used it as a feature extractor for speaker verification and analyzed its performance.

Since LDA training is done in a purely unsupervised way, the topics will contain information regarding many variations in speech in addition to speaker variabilities. This is similar to the total variability space learned by the i-vector models. The fixed size and low dimensionality of the i-vector representation gives us the opportunity to apply proven supervised pattern recognition techniques such as probabilistic linear discriminant analysis (PLDA). Usually prior to PLDA modeling i-vectors are whitened, length normalized and possibly dimensionality reduced to better conform to the assumptions of the PLDA model such as Gaussianity and unimodality [11]. Since this process is the state-of-the-art method to extract speaker and model specific information within the i-vectors, we will use the same procedure to model LDA based topic probability feature vectors.

To use LDA as a feature extractor, we first obtain the word posterior counts using our acoustic vocabulary. Then LDA model is used to extract the expected value of topic distribution vector $\theta$ for each utterance and take its logarithm to better fit the PLDA Gaussianity assumptions. The log topic distribution vectors are our new features. These features are whitened and projected to a lower subspace using linear discriminant analysis. Two-covariance PLDA models [12] are trained on this subspace and used for log-likelihood ratio scoring. When a target speaker has many utterances we take the average of the feature vectors before two-covariance scoring as usually done in i-vector-PLDA systems.

# 5. Experiments

## 5.1. Datasets

The proposed LDA front-end and a baseline i-vector-PLDA system are tested on NIST SRE12 [13] dataset. NIST provided lists of speech segments belonging to each of the 1918 SRE12 target speakers. These training speech segments are from SRE06, SRE08, and SRE10 corpora. The I4U consortium, one of the participants of NIST SRE12, divided these training segments into two speaker verification tasks *Dev* and *Eval*. Each task is composed of two lists : *Train* and *Test*. The utterances in Dev-Test and Eval-Test are non-overlapping. The Dev-Train and Dev-Test utterances are included in Eval-Train. The training and test utterances in the lists have different Lingustic Data Consortium labels. For each segment two noisy versions are generated, one having 6 dB SNR and the other having 15 dB SNR. We have used 10 HVAC noise files downloaded from the internet and crowd noise files generated by summing several hundreds of utterances from NIST SRE corpora to generate noisy versions of the dataset. From each segment and its noisy version, we have also generated truncated versions by randomly taking portions of length between 20s -160s. More detailed information about I4U development list can be found in [14].

## 5.2. i-vector-PLDA Baseline System

We have used MFCC features of 39 dimension containing 19 static, 19 delta and 1 delta-energy coefficients. We have used an energy based bi-Gaussian classifier for voice activity detection. Feature warping with a 3s window is applied to MFCC features after voice activity detection. 2048 mixture gender independent UBM is trained using segments from NIST SRE04, SRE05 corpora as well as segments from Dev-Train list. Noisy and truncated segments are not used in UBM training. Same utterances are used for training gender-dependent i-vector models with i-vector dimension set to 600. Linear Discriminant Analysis is used to further reduce the dimension to 200. The i-vectors are then length normalized, and used to train gender dependent Two-Covariance PLDA models. In training linear discriminant analysis and two-covariance models noisy and truncated versions of the original utterances are also used. When the target speaker has multiple training segments, the average of the training i-vectors are used for training the target speaker model. Znorm score normalisation is used in all experiments. We have used the Bosaris toolkit [15] for fusion and calibration.

## 5.3. LDA front-end

We have used the same UBM and gender-dependent i-vector models for acoustic vocabulary construction. We have tested three different acoustic vocabulary sizes; 2048, 10240, and 20480. For each vocabulary size, we have tested three values for the number of topics K; 200, 400, and 600. In training LDA models, we have used symmetric Dirichlet priors and set $\alpha = 50/K$ and $\eta = 0.01$ as suggested by [2]. The same segments used in UBM and i-vector training are used in gender-

dependent LDA model training. The log of topic probability vectors generated using LDA models are used to train whitening transforms. During whitening, we have reduced the dimensionality when appropriate. After whitening, we have trained linear discriminate analysis models. After linear discriminant analysis projection, the size of the log topic probability vectors are reduced to 300 for models with number of topics K=400, and to 400 when K=600. For models with 200 topics, we have not applied any dimensionality reduction. Finally, two covariance models are trained using the same data as in the i-vector-PLDA system. When the target speaker has multiple training segments, averaging is performed.

### 5.4. Results

The *Dev* task lists from I4U is used for analyzing the effect of vocabulary size and number of topics for speaker verification performance of the proposed system. We have used equal error rate (EER), minimum value of decision cost function of NIST SRE08 evaluation (DCF08), and minimum value of decision cost function of NIST SRE10 evaluation (DCF10) as our performance metrics. Results for male speakers on clean, and noisy segments are shown in Table 1 and Table 2, respectively. Results for female speakers on clean, and noisy segments are shown in Table 3 and Table 4, respectively. No calibration is performed for these experiments. The best results are obtained when vocabulary size and number of topics are set to 10240 and 600, respectively.

We have further tested the LDA system on NIST SRE12 extended test condition. NIST determined 5 subsets of this test condition as common conditions. In all of the five common conditions, all segments of the target speakers are used for training. For test segments:

- Common condition 1 (CC1) involves trials with interview speech without added noise,
- Common condition 2 (CC2) involves trials with telephone channel speech without added noise,
- Common condition 3 (CC3) involves trials with interview speech with added noise,
- Common condition 4 (CC4) involves trials with telephone channel speech with added noise.
- Common condition 5 (CC5) involves trials with telephone channel speech intentionally collected in a noisy environment.

Note that data in common conditions differ from each other in factors not mentioned in the definitions of the common conditions. For example, clean test data are shorter on average than noise added data. So comparing directly the results of two separate common conditions is not meaningful. Additional information about NIST SRE12 Evaluation can be found in [13].

We evaluated the performance of the LDA speaker verification system on the first four common conditions, and compared with the baseline i-vector-PLDA system described in section 5.2. We have set the vocabulary size and number of topics to their best performing values, namely 10240 and 600. We have calibrated the systems using linear logistic regression with the Bosaris toolkit. During calibration we have also used the logarithms of the amount of speech frames (after VAD) in the test segment and average amount of speech frames in the training segments of the target speaker as quality factors. We have used NIST SRE10 DCF point ($C_{Miss} = 1$, $C_{FA} = 1$, $P_{target} = 0.001$) in calibration. We have used the scores obtained in Dev-task of I4U lists for training linear logistic regression. We have also fused the scores of the two systems in order to determine if the LDA system gives additional benefit over the i-vector-PLDA system. We assumed the probability of known non-targets equal to 0. We have used equal error rate, minimum value of the normalized DCF (minDCF), and actual value of the normalized DCF (actDCF) as our performance metrics. The results are summarised in Table 5.

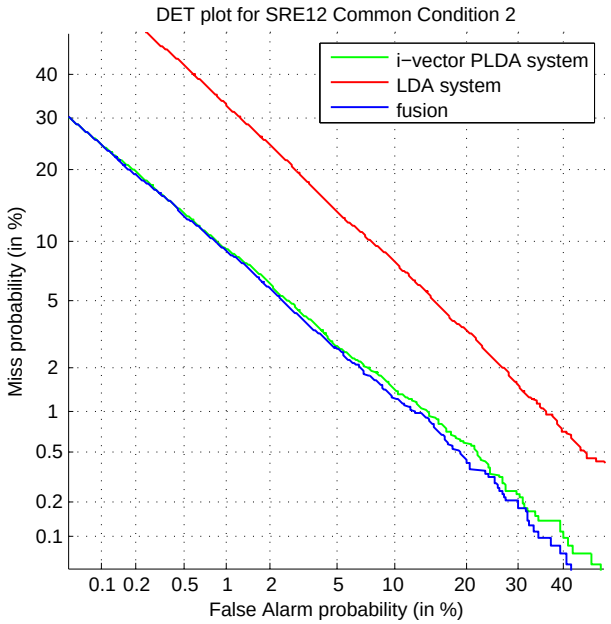


Figure 3: *DET curve for ivector-PLDA, LDA and fused systems for common condition 2 of NIST SRE12 corpus.*

The i-vector-PLDA system clearly outperforms the LDA system in all common conditions. However, performance improves nearly for all common conditions and evaluation metrics when fused with the LDA system. Note that the individual systems are also calibrated with the same quality factors using the Bosaris toolkit, so the improvement comes solely from the fusion step. The DET curves for all the three systems are given in Figure 3 for common condition 2, and in Figure 4 for common condition 4. The improvement obtained by fusing the two systems, although being slight, can be seen in a wide range of operating points especially in the low miss probability region.

## 6. Conclusion

We have proposed a new feature extractor for speaker verification utilizing latent Dirichlet allocation, one of the most popular topic modelling methods in the literature. Instead of transcribing the speech data prior to LDA, we have used a variant of LDA that works directly on continuous descriptor vectors. We have proposed a method to construct a two-level hierarchical acoustic vocabulary to efficiently calculate posterior counts of acoustic words even for moderate sizes of the vocabulary. We have presented a speaker verification system taking log-topic probability vectors as input and using two-covariance PLDA model for log-likelihood ratio scoring. Optimal values of the vocabulary size and number of topics are investigated. The proposed system gave encouraging results on NIST SRE corpora, while still performing worse than a state-of-the-art i-vector-PLDA system.

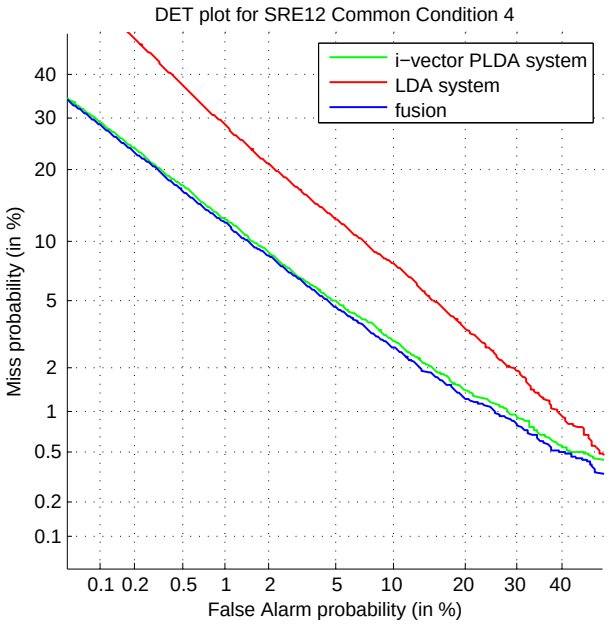


Figure 4: *DET curve for ivector-PLDA, LDA and fused systems for common condition 4 of NIST SRE12 corpus.*

Fusing the scores of the two systems improved the performance of the baseline in NIST SRE12 common conditions, suggesting the log-topic probability vectors having additional information to the i-vector.

We are planning to experiment more with the model to understand what the LDA topics are modeling for speech data. Another direction of research may be working on discriminative variants of LDA model, such as maximum margin topic models [16]. Training LDA using speaker labels and trying to find topics best discriminating speakers may improve the performance of the proposed system.

## 7. References


[1] David M Blei, Andrew Y Ng, and Michael I Jordan, "Latent dirichlet allocation," *the Journal of machine Learning research*, vol. 3, pp. 993–1022, 2003.

[2] Mark Steyvers and Tom Griffiths, "Probabilistic topic models," *Handbook of latent semantic analysis*, vol. 427, no. 7, pp. 424–440, 2007.

[3] Daphna Weinshall, Dmitri Hanukaev, and Gal Levi, "LDA topic model with soft assignment of descriptors to words," in *Proc. of the 30th International Conference on Machine Learning (ICML-13)*, 2013.

[4] Diane Larlus and Frédéric Jurie, "Latent mixture vocabularies for object categorization and segmentation," *Image and Vision Computing*, vol. 27, no. 5, pp. 523–534, 2009.

[5] Kong-Aik Lee, Chang Huai You, Ville Hautamäki, Anthony Larcher, and Haizhou Li, "Spoken language recognition in the latent topic simplex," in *Proc. Interspeech*, 2011, pp. 2933–2936.

[6] Shan Jin, Hemant Misra, Thomas Sikora, and Joemon Jose, "Automatic topic detection strategy for information retrieval in spoken document," in *Image Analysis for Multimedia Interactive Services, 2009. WIAMIS'09. 10th Workshop on*. IEEE, 2009, pp. 300–303.

[7] Timothy J Hazen, "Direct and latent modeling techniques for computing spoken document similarity," in *Proc. Spoken Language Technology Workshop (SLT)*. IEEE, 2010, pp. 366–371.

[8] Hanna M Wallach, David M Mimno, and Andrew McCallum, "Rethinking LDA: Why priors matter.," in *NIPS*, 2009, vol. 22, pp. 1973–1981.

[9] Thomas L Griffiths and Mark Steyvers, "Finding scientific topics," *Proc. of the National academy of Sciences of the United States of America*, vol. 101, no. Suppl 1, pp. 5228–5235, 2004.

[10] Patrick Kenny, "A small footprint i-vector extractor," in *Odyssey 2012-The Speaker and Language Recognition Workshop*, 2012.

[11] Daniel Garcia-Romero and Carol Y Espy-Wilson, "Analysis of i-vector length normalization in speaker recognition systems.," in *Proc. Interspeech*, 2011, pp. 249–252.

[12] Niko Brümmer and Edward De Villiers, "The speaker partitioning problem," in *Proceedings of the Odyssey Speaker and Language Recognition Workshop, Brno, Czech Republic*, 2010.

[13] Craig S Greenberg, Vincent M Stanford, Alvin F Martin, Meghana Yadagiri, George R Doddington, John J Godfrey, and Jaime Hernandez-Cordero, "The 2012 NIST speaker recognition evaluation," in *Proc. Interspeech*, 2013.

[14] R Saeidi, KA Lee, T Kinnunen, T Hasan, B Fauve, PM Bousquet, E Khoury, PL Sordo Martinez, JMK Kua, CH You, et al., "I4U submission to NIST SRE 2012: A largescale collaborative effort for noise-robust speaker verification," in *Proc. Interspeech*, 2013.

[15] Niko Brümmer and Edward de Villiers, "The bosaris toolkit: Theory, algorithms and code for surviving the new dcf," in *NIST SRE Analysis Workshop*, 2011.

[16] Jun Zhu, Amr Ahmed, and Eric P Xing, "MedLDA: maximum margin supervised topic models for regression and classification," in *Proc. of the 26th Annual International Conference on Machine Learning*. ACM, 2009, pp. 1257–1264.

Table 1: *Results of the system for male clean data condition for various vocabulary sizes and number of topics K .*

| Vocabulary Size | K=200 | | | K=400 | | | K=600 | | |
|---|---|---|---|---|---|---|---|---|---|
| | EER | DCF08 | DCF10 | EER | DCF08 | DCF10 | EER | DCF08 | DCF10 |
| 2048 | 1.83 | 0.0998 | 0.4914 | 1.62 | 0.0882 | 0.4209 | 1.38 | 0.0722 | 0.3973 |
| 10240 | 1.48 | 0.0825 | 0.4050 | **1.17** | 0.0652 | 0.3416 | 1.21 | **0.0605** | **0.3323** |
| 20480 | 1.59 | 0.082 | 0.4234 | 1.28 | 0.0685 | 0.3838 | 1.25 | 0.0706 | 0.3735 |

Table 2: *Results of the system for male noisy data condition for various vocabulary sizes and number of topics K .*

| Vocabulary Size | K=200 | | | K=400 | | | K=600 | | |
|---|---|---|---|---|---|---|---|---|---|
| | EER | DCF08 | DCF10 | EER | DCF08 | DCF10 | EER | DCF08 | DCF10 |
| 2048 | 3.57 | 0.1871 | 0.6909 | 3.18 | 0.1587 | 0.6146 | 2.83 | 0.1457 | 0.5816 |
| 10240 | 3.11 | 0.1626 | 0.6109 | 2.63 | 0.1392 | 0.5434 | **2.44** | **0.1313** | **0.5231** |
| 20480 | 3.19 | 0.1660 | 0.6380 | 2.91 | 0.1476 | 0.5828 | 2.77 | 0.1494 | 0.5909 |

Table 3: *Results of the system for female clean data condition for various vocabulary sizes and number of topics K .*

| Vocabulary Size | K=200 | | | K=400 | | | K=600 | | |
|---|---|---|---|---|---|---|---|---|---|
| | EER | DCF08 | DCF10 | EER | DCF08 | DCF10 | EER | DCF08 | DCF10 |
| 2048 | 3.4 | 0.1827 | 0.6814 | 2.88 | 0.1535 | 0.6017 | 2.55 | 0.1345 | 0.5676 |
| 10240 | 2.85 | 0.1567 | 0.6536 | 2.49 | 0.1285 | **0.5641** | **2.36** | **0.1217** | 0.5688 |
| 20480 | 3.05 | 0.1642 | 0.6643 | 2.87 | 0.1446 | 0.6411 | 2.75 | 0.1445 | 0.6410 |

Table 4: *Results of the system for female noisy data condition for various vocabulary sizes and number of topics K .*

| Vocabulary Size | K=200 | | | K=400 | | | K=600 | | |
|---|---|---|---|---|---|---|---|---|---|
| | EER | DCF08 | DCF10 | EER | DCF08 | DCF10 | EER | DCF08 | DCF10 |
| 2048 | 5.39 | 0.2943 | 0.8373 | 4.72 | 0.2438 | 0.7781 | 4.36 | 0.2258 | 0.7457 |
| 10240 | 4.79 | 0.2617 | 0.8188 | 4.30 | 0.2218 | 0.7430 | **4.08** | **0.2103** | **0.7320** |
| 20480 | 4.75 | 0.2647 | 0.8296 | 4.64 | 0.2455 | 0.7961 | 4.50 | 0.2391 | 0.8086 |

Table 5: *Results of the system for NIST SRE12 extended task common conditions.*

| System | CC1 | | | CC2 | | | CC3 | | | CC4 | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | EER | minDCF | actDCF | EER | minDCF | actDCF | EER | minDCF | actDCF | EER | minDCF | actDCF |
| i-vector-PLDA | 4.44 | 0.5005 | 0.5520 | 3.69 | 0.5509 | 0.6508 | 3.25 | 0.3941 | 0.4229 | 4.91 | 0.5984 | 0.6883 |
| LDA front-end | 8.36 | 0.7687 | 0.7839 | 8.82 | 0.9067 | 0.907 | 5.15 | 0.6435 | 0.6518 | 8.53 | 0.916 | 0.9202 |
| fused | **4.16** | **0.4962** | **0.5452** | **3.51** | 0.5563 | 0.6508 | **3.09** | **0.3895** | **0.41** | **4.73** | **0.5924** | **0.6838** |