SIMUni: Sampling Impostors from Misfit Universal Background Models in accelerometric gait biometric verification

Vinay Uday Prabhu
UnifyID Labs
UnifyID Inc
Redwood City, CA 94063
vinay@unify.id

Daniel Wen
CS, Carnegie Mellon University
UnifyID Inc
Redwood City, CA 94063
daniel@cmu.edu

John Whaley
UnifyID Labs
UnifyID Inc
Redwood City, CA 94063
john@unify.id

Abstract

In this paper, we would like to disseminate surprisingly positive results we obtained by using a framework for generating impostor features in the context of training user-specific models in accelerometric gait biometric user verification. We propose that we directly sample from a *poorly fit* Universal Background - Gaussian Mixture Model (UBM-GMM) to generative negative class features, which on the face of it, seems like an unreasonable proposal, and combining these with the positive class user-enrollment features to train local user-specific shallow classifiers. Through empirical analysis on the state-of-the-art dataset, we showcase that this simple approach outperforms the classical UBM-GMM approach with or without score normalization, a result that was rather unexpected.

1 Introduction

In figure 1, we see the system diagram for the classical Universal Background Model - Gaussian Mixture Model (UBM-GMM) approach, which is nearly two decades old (See Reynolds et al. (2000)), but still finds widespread usage in the biometrics community as evinced by its recent usage in Arabic dialect identification (Moftah et al. (2018)), mobile phone clustering from speech recordings (Li et al. (2018)) and Human identification by sensor kinematics (Neverova et al. (2016)).
The system entails three phases.
1: The offline UBM development phase: We begin with a vast corpus of raw user data, $\mathcal{X}_{UBM} = \{\mathbf{x} \in \mathbb{R}^{n_d}\}$ and a feature extractor, $\xi() : \mathbb{R}^{n_d} \rightarrow \mathbb{R}^{n_f}$, to build a universal background feature dataset, $\mathcal{F}_{UBM} = \{\mathbf{f} : \mathbf{f} = \xi(\mathbf{x}) \forall \mathbf{x} \in \mathcal{X}_{UBM}\}$. Now, using model learning algorithms such as Expectation-Maximization (McLachlan et al. (2004)), we fit a simple GMM generative model to this dataset to obatin,

$$\begin{aligned} p_{\Theta_{UBM}}(\mathbf{f}) &= \sum_{m=1}^{M} \pi_m N(\mu_m, \Sigma_m), where, \\ \Theta_{UBM} &:= (\mathbf{\Pi}_{UBM}, \mathrm{M}_{UBM}, \mathbf{\Sigma}_{UBM}) = \{(\pi_m, \mu_m, \Sigma_m)\}_{m=1}^{M} \end{aligned} \tag{1}$$

2: User enrollment phase using MAP adaptation: Given a small amount of user enrollment data with n_E enrollment feature samples (constituting $\mathcal{F}_{user} \in \mathbb{R}^{n_f \times n_E}$, a user specific GMM model is learned using the UBM-GMM in eq.1 as the prior using a *MAP-mean-adaptation* procedure (see Reynolds et al. (2000)), to get,

$$p_{\Theta_{user}}(\mathbf{f}) = \sum_{m=1}^{M} \pi_m N\left(\mu_m^{(MAP)}, \Sigma_m\right) \tag{2}$$

BayLearn 2018 (2018), Menlo Park, CA, USA.

3: Real world test phase: During the test phase, in response to an incoming feature vector $\mathbf{f}_{test}$, we compute the log-likelihood function $\Lambda(\mathbf{f})$, as,

$$\Lambda(\mathbf{f}_{test}) = \log p(\mathbf{f}_{test}; \Theta_{user}) - \log p(\mathbf{f}_{test}; \Theta_{UBM}). \tag{3}$$

and perform the following log-likelihood ratio (LLR) test:

$$\Lambda\left(\mathbf{f}_{test}\right) \begin{cases} \geq \tau_{user} : User\ chosen \\ < \tau_{user} : Universe \end{cases} \tag{4}$$

Here τ_{user} is a user-specific threshold. Often, instead of using this *raw* LLR, some form of score normalization usually is performed before the thresholding. In this paper, we tried multiple score normalization techniques and the test-score normalization (*T-norm*) technique[1] of Auckenthaler et al. (2000), where the normalized LLR is simply $\left(\frac{\Lambda(\mathbf{f})-\mu_T}{\sigma_T}\right)$ gave the best result.

2 SIMUni framework

In this section, we propose the SIMUNi framework. The idea is as detailed in fig 2.
After the UBM-GMM model is leaned, during the enrollment phase, we form a *positive class* feature-set, $F^{+}_{user} = \{\xi\left(\mathbf{x}_{user,i}\right); i = 1, ..., n_E\}$, and then sample n_E features from the UBM-GMM model (in eq1) to obtain the corresponding *negative class* feature-set, $F^{-}_{user} = \{\mathbf{f}_{user,i} \sim p_{\Theta_{UBM}}\left(\mathbf{f}\right); i = 1, ..., n_E\}$. Now, we use the positive and negative class feature sets thus obtained to train a user-specific discriminative binary classifier, $\zeta_{\psi_{user}}\left(\mathbf{f}\right) : F \rightarrow \{-1, +1\}$. In the case of the ν -Support Vector Classifier with RBF kernel[2], we have, $\psi_{user} = (\nu_{user}, \gamma_{user}, C_{user})$. Now, during the test phase in lieu of performing the LLR test (as in eq.5) , we pass the test feature vector through the learned user-specific classifier to perform user-verification, as,

$$\zeta_{\psi_{user}}\left(\mathbf{f}_{test}\right) \begin{cases} = +1 : User\ chosen \\ = -1 : Universe \end{cases} \tag{5}$$

3 Experiments and Results

We trained a 128-component 80-dimensional UBM-GMM with diagonal matrices on a 1.2 million accelerometric *GaitNet* gait-cycle dataset where each gait-cycle was of dimension 4×100 (The 4 rows mapped to the x, y, z and mag sensor axes and 100 was the temporal span after re-sampling). The features were extracted by passing each of the gait-cycle matrices through a Deep CNN feature extractor. This deep feature extractor was constructed by slicing the *UnifyIDNet* [3] before the *soft-max* layer, yielding a 80-dimensional feature vector per gait-cycle. The test dataset consisted of 155 users, each with 1000 gait-cycles. A $400 - 100 - 500$ train-validate-test split was deployed and a random attacker model was used where 500 negative-class test gait-cycles not belonging to the user were randomly sampled from the dataset of 154 remaining users to obtain the classification results.
In fig 3, we see the Detection Error Trade-off (DET) curves comparing the traditional UBM-GMM approach with the proposed SIMUni framework. `UBM` refers to the case of the classical UBM-GMM approach, `SVM` refers to the SIMUNi framework scenario where the per-user classifier was an SVM with RBF kernel (as detailed in section-2) and `RF` refers to the SIMUNi framework scenario where the per-user classifier was a Random-Forest. `(T)` in the legend refers to the cases where the above mentioned algorithms was used in conjunction with *T-norm* score normalization procedure. The input into the T-norm computation was the LLR (as in (3)) for UBM. For RF and SVM, we used the output of the `predict_log_proba` methods implemented in `scikit-learn`(Pedregosa et al. (2011)). As seen, the EER (Equal Error Rate) for the SIMUni classifiers is better than the UBM-GMM approach with or without score-normalization. In fig.4, we see the t-SNE plot for a random user showcasing the training and testing deep feature vector embeddings and the resultant positive and negative class support vectors upon using the SVM-RBF classifier. This is to showcase how surprisingly *reasonable* the sampled negative class features were in terms of being able to *cover* the adversary space.

[1] The procedure for obtaining the T-norm mean correction (μ_T) and scale (σ_T) paramaters is pretty standard and left out for brevity

[2] `http://scikit-learn.org/stable/modules/generated/sklearn.svm.NuSVC.html`

[3] that was trained on the GaitNet dataset in a discriminative 1300 class setting

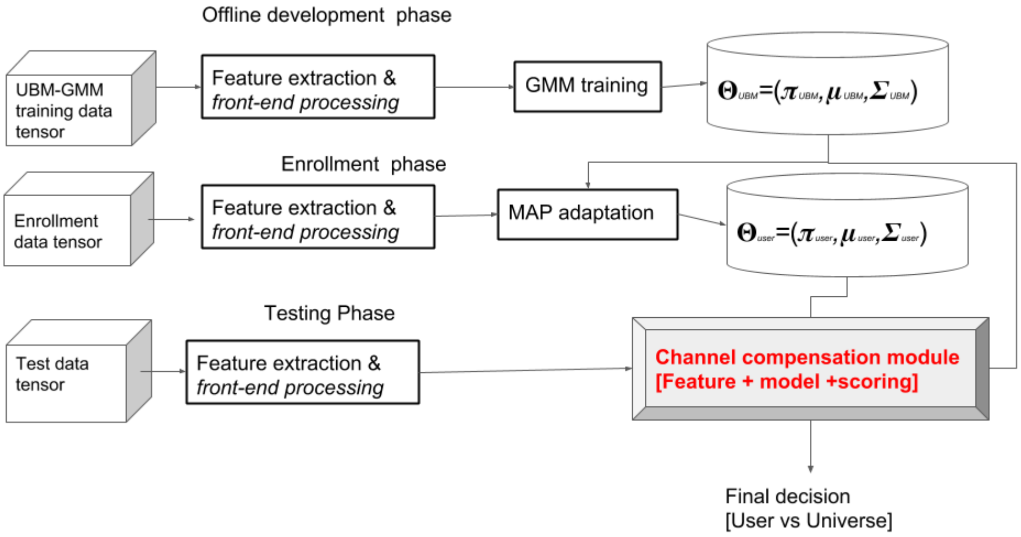
Offline development phase
UBM-GMM training data tensor
Feature extraction & *front-end processing*
GMM training
$\Theta_{UBM}=(\pi_{UBM},\mu_{UBM},\Sigma_{UBM})$
Enrollment phase
Enrollment data tensor
Feature extraction & *front-end processing*
MAP adaptation
$\Theta_{user}=(\pi_{user},\mu_{user},\Sigma_{user})$
Testing Phase
Test data tensor
Feature extraction & *front-end processing*
Channel compensation module [Feature + model +scoring]
Final decision [User vs Universe]

Figure 1: System diagram explaining the *UBM-GMM* framework

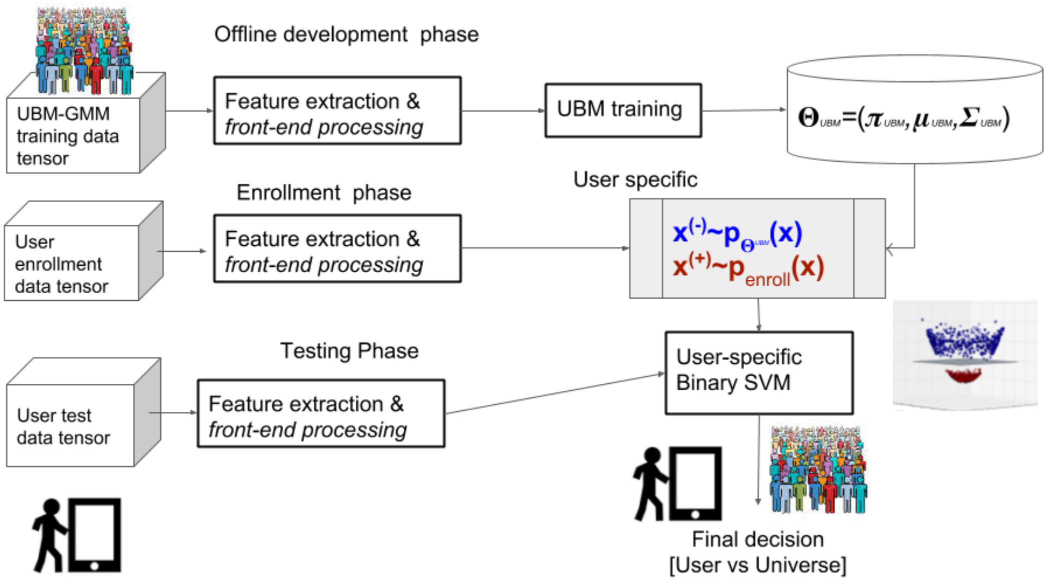
Offline development phase
UBM-GMM training data tensor
Feature extraction & *front-end processing*
UBM training
$\Theta_{UBM}=(\pi_{UBM},\mu_{UBM},\Sigma_{UBM})$
Enrollment phase
User specific
User enrollment data tensor
Feature extraction & *front-end processing*
$x^{(-)}\sim p_{\Theta_{UBM}}(x)$
$x^{(+)}\sim p_{enroll}(x)$
Testing Phase
User test data tensor
Feature extraction & *front-end processing*
User-specific Binary SVM
Final decision [User vs Universe]

Figure 2: System diagram explaining the *SIMUNi* framework

References

Auckenthaler, Roland, Carey, Michael, and Lloyd-Thomas, Harvey. Score normalization for text-independent speaker verification systems. *Digital Signal Processing*, 10(1-3):42–54, 2000.

Li, Yanxiong, Zhang, Xue, Li, Xianku, Zhang, Yuhan, Yang, Jichen, and He, Qianhua. Mobile phone clustering from speech recordings using deep representation and spectral clustering. *IEEE Transactions on Information Forensics and Security*, 13(4):965–977, 2018.

McLachlan, Geoffrey J, Krishnan, Thriyambakam, and Ng, See Ket. The em algorithm. Technical report, Papers/Humboldt-Universität Berlin, Center for Applied Statistics and Economics (CASE), 2004.

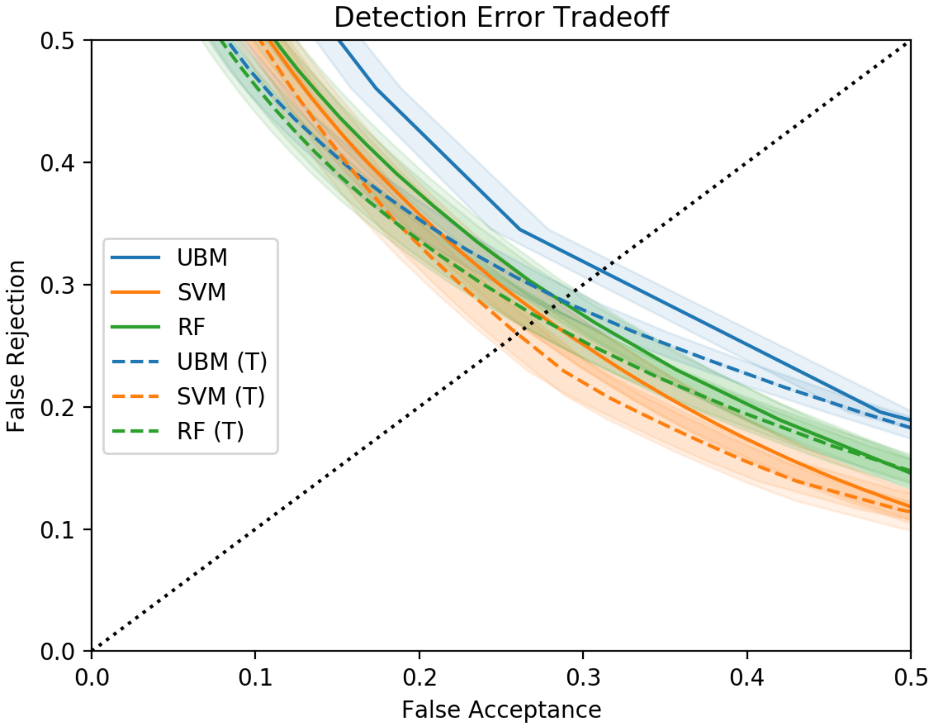

Figure 3: DET curve comparisons

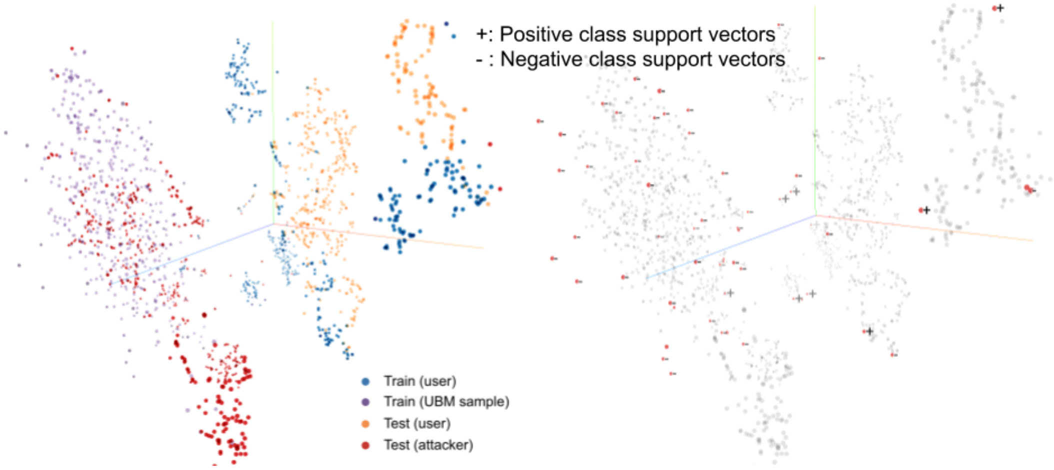

Figure 4: t-SNE plot for a random user's features

Moftah, Mohsen, Fakhr, Mohammed Waleed, and El Ramly, Salwa. Arabic dialect identification based on motif discovery using gmm-ubm with different motif lengths. In *Natural Language and Speech Processing (ICNLSP), 2018 2nd International Conference on*, pp. 1–6. IEEE, 2018.

Neverova, Natalia, Wolf, Christian, Lacey, Griffin, Fridman, Lex, Chandra, Deepak, Barbello, Brandon, and Taylor, Graham. Learning human identity from motion patterns. *IEEE Access*, 4: 1810–1820, 2016.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

Reynolds, Douglas A, Quatieri, Thomas F, and Dunn, Robert B. Speaker verification using adapted gaussian mixture models. *Digital signal processing*, 10(1-3):19–41, 2000.