Toward Accurate and Efficient Feature Selection for Speaker Recognition on Wearables

Rui Liu
Dept. of Computer Science
Dartmouth College
rliu@cs.dartmouth.edu

Reza Rawassizadeh
Dept. of Computer Science
Dartmouth College
rrawassizadeh@acm.org

David Kotz
Dept. of Computer Science
Dartmouth College
kotz@cs.dartmouth.edu

ABSTRACT

Due to the user-interface limitations of wearable devices, voice-based interfaces are becoming more common; speaker recognition may then address the authentication requirements of wearable applications. Wearable devices have small form factor, limited energy budget and limited computational capacity. In this paper, we examine the challenge of computing speaker recognition on small wearable platforms, and specifically, reducing resource use (energy use, response time) by trimming the input through careful feature selections. For our experiments, we analyze four different feature-selection algorithms and three different feature sets for speaker identification and speaker verification. Our results show that Principal Component Analysis (PCA) with frequency-domain features had the highest accuracy, Pearson Correlation (PC) with time-domain features had the lowest energy use, and recursive feature elimination (RFE) with frequency-domain features had the least latency. Our results can guide developers to choose feature sets and configurations for speaker-authentication algorithms on wearable platforms.

1. INTRODUCTION

Wearable devices (a.k.a. wearables) have become common, including smartwatches, virtual-reality headsets, body cameras, and smart clothing. Wearables are usually associated with a limited graphical user interface (GUI), or no GUI at all. Therefore, manufacturers and developers are seeking viable options to enable users' interaction with their devices. One of the emerging alternatives is a voice user interface (VUI) [5], because it is feasible to embed a small microphone in many wearables.

In many applications, wearables provide sensitive services, such as monitoring the user's health information, managing the user's calendar, collecting a life-log of photographs, or paying for purchases. Thus, authentication is necessary on wearable devices. Systems that support a keypad, keyboard or GUI often use a password or pincode to authenticate users. Due to the lack of such interfaces in many wearable devices, the authentication process could be performed through a VUI. Some wearables devices are personal, and the device should verify that the wearer is indeed the device owner. Other wearable devices are shared, such as a VR headset used by a household, and the device may need to identify its current user to personalize the experience. *Verification* confirms whether the wearer of a personal device is the expected user: a binary classification problem. On the other hand, *identification* strives to identify the wearer from a group of known users: a multi-class classification problem. In this paper, *speaker authentication* refers to either speaker verification or speaker identification.

Speaker authentication is available on capable computing devices but there are challenges to enabling these voice-based authentication methods on wearable devices. Speaker authentication algorithms are resource intensive [19, 9], but wearables have relatively limited energy and computational capacity [15].

Due to their limited capacity, many wearables off-load computation to a smartphone or to a cloud [8]. This approach may increase response time (latency) and battery utilization, and reduces availability when the smartphone or network is unavailable [16]. Therefore, recent research aims to develop lightweight algorithms that perform machine learning on the device itself [1, 11, 12, 14, 16].

Although these are promising efforts in the design and optimization of wearable machine-learning algorithms, one of the most effective ways to reduce resource usage while retaining accuracy, for any algorithm, is to *reduce the size of the input*. In this paper, we refine input features of the speaker-authentication algorithm toward optimizing the resource efficiency of the algorithm, while maintaining its accuracy. We analyze different feature sets and report their impact on accuracy and resource efficiency, including response time and energy use, for a speaker-authentication algorithm [5]. There are two broad categories of feature sets relevant for speaker-authentication algorithms: time-domain features and frequency-domain features. We analyze both categories together and separately, for both speaker identification and speaker verification.

With our results, developers could choose feature sets and feature-selection algorithms based on their needs for accuracy, energy use, or latency.

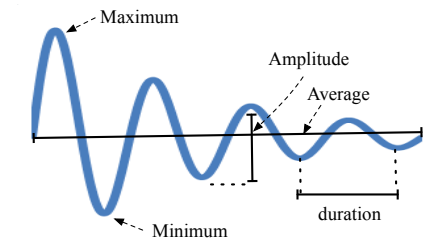
2. METHOD

In this section, we describe the process of extracting features from our datasets. Then, we briefly describe the al-

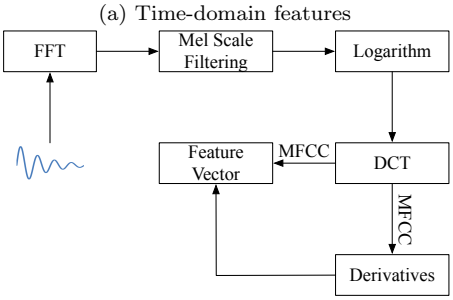
WearSys'17, June 19, 2017, Niagara Falls, NY, USA

ISBN 978-1-4503-4959-8/17/06…$15.00

DOI: http://dx.doi.org/10.1145/3089351.3089352



(a) Time-domain features



(b) Frequency-domain features

Figure 1: Time-domain and frequency-domain features.

gorithms for speaker authentication on a wearable device. Afterward, we describe our feature-selection methods.

2.1 Feature extraction

Audio signal-analysis methods typically focus on two categories of features: time-domain features and frequency-domain features.

Time-domain features are extracted directly from the input signal, typically computed for each window of time within the relevant period. Figure 1(a) shows four common features: amplitude, average, maximum, minimum. We use 12 time-domain features, including the average, min, max, absolute max, zero-cross rate (ZCR), mean-cross rate (MCR), root-mean-square (RMS) energy, logarithm energy, along with the derivatives and second derivatives of the energy values.

Frequency-domain features are extracted by applying a Fourier transform on the input signal to convert the time domain into a frequency domain. From the Fourier coefficients, frequency-domain features can be computed. Mel-Frequency Cepstral Coefficients (MFCCs) are common features for processing voice data [5, 17]. We use 13 MFCCs and their 13 derivatives as frequency-domain features. Figure 1(b) depicts the process of extracting MFCCs and derivatives of the MFCCs.

In this paper, we evaluate feature-selection methods using feature vectors of time-domain features (TD), frequency-domain features (FD), and both (TFD).

2.2 Gaussian Mixture Model algorithm

We use a Gaussian Mixture Model (GMM), which is known to be one of the most effective models for speaker identification and verification [17]. This approach models the distribution of observations using a weighted linear combination of Gaussian densities, where each Gaussian density is parameterized by a mean vector and covariance matrix. Therefore, in this paper, we use GMMs to model the distribution of feature vectors, including time-domain and frequency-domain features, for a given speaker.

To learn the underlying distribution of feature vectors, we use the Expectation-Maximization (EM) algorithm [7] to iteratively refine the mixture of Gaussian densities until the maximum likelihood remains stable. Modeling the covariance matrix in full is computationally expensive; to be able to compute it on a wearable device, we use diagonal covariance matrices because it has been shown that using a larger-dimensional diagonal covariance matrix performs better than a smaller-dimensional full covariance matrix [2]. In this paper, we use 32 Gaussian components for each mixture and all the EM processes terminate within 100 iterations.

Speaker identification: In the training phase, we train a GMM for each subject (i.e., user of the wearable device). In the identification process, feature vectors are extracted for each segment of audio data and given to all the GMMs. Each GMM provides an average log-likelihood of each feature vector belonging to the mixture as output. The identification algorithm then outputs the identity of the subject whose GMM outputs the maximum of average log-likelihoods.

Speaker verification: As above, we train a GMM for each subject; we also learn a threshold for each subject using a 3-fold cross validation on training data. For each fold and subject, we learn a GMM and use the other folds of this subject to learn an average probability (p_1) of correct cases. To learn a threshold for the subject, we use the other subjects' audio data as a training set to learn an average probability (p_2) of incorrect cases. We use the mean $\frac{1}{2}(p_1 + p_2)$ as the threshold for each subject. For verification, the feature vector extracted from a wearer's input audio segment is given to the GMM of the target subject. The GMM outputs a log-likelihood; if this log-likelihood is greater than the threshold, the wearer is accepted. Otherwise, if the log-likelihood is lower than threshold, the wearer is rejected.

2.3 Feature-selection methods

Feature selection is the process of selecting a subset of input variables (features) that are most useful to construct a machine-learning model. The purpose is to simplify the models, reduce input dimensionality, reduce computation, and avoid overfitting. For wearables, a smaller feature vector takes less time to extract, less space to store, and less time to process through a less-complex model. There are three categories of feature-selection methods: filter methods, wrapper methods, and embedded methods [4, 10]. We selected one algorithm in each category: Pearson correlation, Sequential Forward Selection, and Recursive Feature Elimination, respectively [4]. Moreover, Principal Component Analysis (PCA) is a common method to reduce the dimensionality of data and it is common to use it for feature selection [4]. Therefore, we also use the PCA algorithm.

Filter methods: A filter method first ranks the features by some metric, such as the Pearson correlation or Mutual Information. This measure is chosen to capture the usefulness of the feature in describing characteristics of the target. Second, the filter method selects the highest-ranked features as a subset. We use the Pearson correlation, which describes the linear association between two given variables.

Wrapper methods: A wrapper method uses a predictive model to score the feature subsets. It trains a predictive model on each feature subset and tests the model on the dataset. An accuracy metric is used to score each feature subset. We use Sequential Forward Selection (SFS), which starts with an empty feature set and iteratively tests each

possible feature and adds the feature that improves the accuracy of the model best. This process continues until no features can improve the model. We used the Lasso Regressor as a predictor.

Embedded methods: An embedded method incorporates the feature selection as a part of the training process of a supervised estimator. The process discards less-contributing features, such as less-important features in a tree-based estimator, lower-weighted features in a SVM model, or features with smaller coefficients in a linear model. We use Recursive Feature Elimination (RFE) with the Lasso Regressor. We recursively remove the feature that has smallest absolute coefficient.

Principal Component Analysis: PCA is a statistical method used to reduce the dimensionality of data. It converts a set of inputs of possibly correlated variables into a set of values of linearly uncorrelated variables called principal components. The number of principal components is less than or equal to the number of original variables. We learn a PCA model from the feature vectors in a training set and use the model to transform the feature vectors from the testing set.

3. EVALUATION

In this section we introduce the two datasets that we used for our experiments. Then, we evaluate the accuracy of speaker identification/verification based on different combinations of features. Next, we report the (i) energy use and (ii) latency for speaker authentication based on different combinations of feature selections. We conducted our experiments on a Raspberry Pi 3 model B, which has a 1.2 GHz quad-core CPU and 1 GB RAM.[1] Existing wearable devices, such as Huawei Watch, have a similar hardware configuration.[2] To implement the GMM algorithm, we used Python version 3.5.2 with numpy (1.12.1), scipy (0.19) and scikit-learn (0.18.1).

3.1 Datasets

We used two datasets: CHAINS and Vocal Resonance. The CHAINS dataset [6] includes audio from subjects reading aloud the first paragraph of four long passages, including *The Rainbow Text*, *The Cinderella Story*, *The North Wind*, and *The Members of the Body*. This dataset also includes 33 short sentences. Each short sentence includes about 3 seconds of audio. The Vocal Resonance dataset [5] includes audio from subjects reading the entire *The Rainbow Text* and a few paragraphs from *The Wind in the Willows*. In the CHAINS dataset, we used the long passages as training data and short sentences as testing data. In Vocal Resonance dataset, we used *The Rainbow Text* as training data and divided the paragraphs from *The Wind in the Willows* into 3-second segments as testing data. Table 1 lists the number of subjects and the duration of training/testing data partition for each subject.

3.2 Accuracy

Identification algorithms output the identity of a subject. If the output is correct, we say it is a True Positive (TP); otherwise, we have a False Positive (FP). Because identification algorithms do not have negative outputs, there are no True Negative (TN) or False Negative (FN) cases. We use Precision ($\frac{TP}{TP+FP}$) to evaluate the accuracy of identification.

Table 1: Training and testing data in each dataset

Dataset	Num. of Subjects	Duration of Train Dataset	Duration of Test Dataset
CHAINS	36	∼170 sec	33 × ∼3 sec
Vocal Resonance	12	∼107 sec	15 × 3 sec

Verification algorithms output positive (the speaker is the expected subject) or negative (otherwise); we thus have four cases: TP (the algorithm outputs positive and is correct), TN (the algorithm outputs negative and is correct), FN (the algorithm outputs negative and is incorrect), and FP (the algorithm outputs positive and is incorrect). We use Balanced Accuracy ($\frac{1}{2}(\frac{TP}{TP+FP} + \frac{TN}{TN+FN})$) to report the accuracy of verification.

As noted, we used time-domain features (TD), frequency-domain features (FD), and the combination of TD and FD (TFD) as feature sets. In addition, we used four feature-selection methods: PC, SFS, RFE, and PCA.

Figure 2 shows that frequency-domain features outperform time-domain features in accuracy metrics if used separately. In the CHAINS dataset, the highest accuracy metrics using frequency-domain features were 0.997 for identification and 0.998 for verification while the metrics were 0.564 and 0.775 respectively using time-domain features. In the Vocal Resonance dataset, the highest accuracy metrics were 0.952 and 0.963 for identification and 0.784 and 0.886 for verification. All the highest accuracy metrics are achieved with PCA-selected features. However, the speaker-authentication algorithms using the combination of time-domain and frequency-domain features (TFD) overfit when using many features: the accuracy declined as more features were included, given a sequence of TFD features in both datasets and both identification/verification algorithms.

3.3 Efficiency

On wearable devices, efficiency (in energy and time) is critical. To report the efficiency, we chose the feature subsets with the highest accuracy metrics on the preceding experiments. In particular, we used the feature selections marked with vertical dashed lines from Figure 2. We measured energy use and response time by averaging 100 runs for feature extraction and speaker-authentication algorithms.

We measured the energy used for feature extraction and speaker-authentication algorithms over 100 runs with a Monsoon Power Monitor.[3] Likewise, we measured average latency incurred for feature extraction and speaker-authentication algorithms. Table 2 aggregates results of accuracy, energy use, latency, feature-selection algorithm and feature set by highest accuracy, lowest energy use or least latency. Although there was no consistent winner, the table shows that frequency-domain feature subsets with PCA tended to have higher accuracy metrics. In terms of energy, time-domain feature subsets with PC had lower power drain. Feature subsets from frequency-domain features with RFE had less latency.

Figure 3 plots accuracy metrics, energy use, and response time for each feature set selected by feature-selection algo-

[1]https://www.raspberrypi.org/products/raspberry-pi-3-model-b

[2]https://en.wikipedia.org/wiki/Huawei_Watch

[3]https://www.msoon.com/LabEquipment/PowerMonitor/

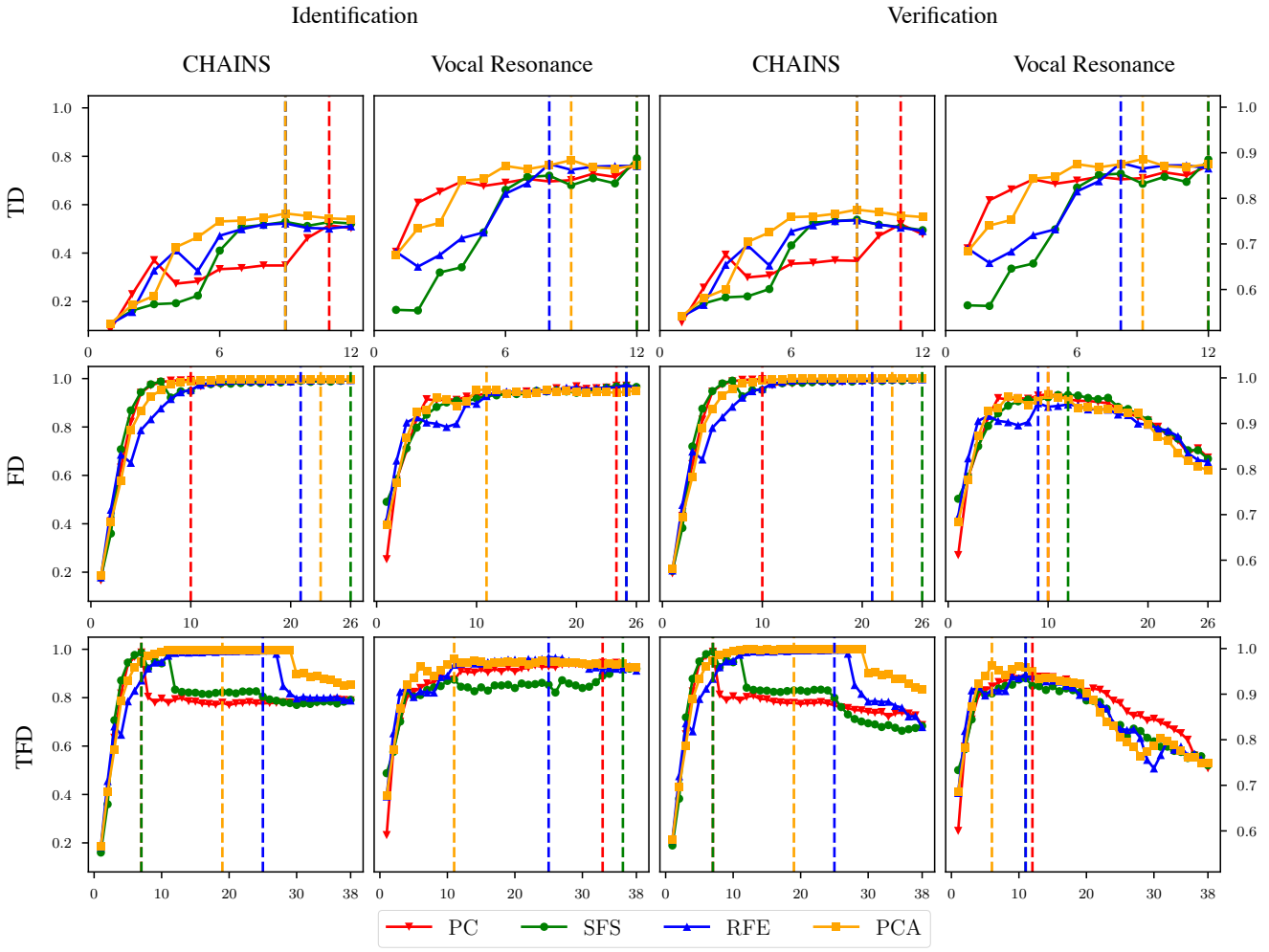


Figure 2: The accuracy metrics for each dataset, feature set and number of features selected. The x-axis represents the number of features in each feature set: 12 is the maximum number of features in TD, 26 is the maximum number in FD, and 38 is the maximum number in TFD. The y-axis represents the Precision for identification and Balanced Accuracy for verification (note identification and verification have different y scales). Feature selections with the highest accuracy metrics are marked with vertical dashed lines for each dataset and feature set (best viewed in color).

Table 2: Aggregated results by highest accuracy, lowest energy (mW) and least latency (second).

		CHAINS		Vocal Resonance	
		Identification	Verification	Identification	Verification
Highest Accuracy	Accuracy	0.997	0.999	0.963	0.963
	Energy	266.59	317.03	281.21	305.03
	Latency	1.897	1.138	1.657	1.138
	Algorithm	PCA	PCA	PCA	PCA
	Featureset	FD	FD	TFD	FD
Lowest Energy	Accuracy	0.995	0.743	0.771	0.871
	Energy	263.05	263.58	274.7	293.68
	Latency	1.945	1.547	1.386	0.974
	Algorithm	RFE	PC	PC	PC
	Featureset	TFD	TD	TD	TD
Least Latency	Accuracy	0.987	0.997	0.947	0.939
	Energy	265.79	300.98	275.38	300.03
	Latency	1.473	0.970	1.334	0.972
	Algorithm	SFS	RFE	RFE	RFE
	Featureset	TFD	FD	FD	FD

rithms on both datasets. This figure shows that FD and TFD were usually more accurate than TD with no significant energy or latency differences.

3.4 Limitations

Although we analyzed different combinations of feature sets and feature-selection algorithms, our approach has several limitations.

First, we evaluated only one machine-learning algorithm (GMM); although it is known to be used on wearable devices for speaker identification/verification [5], we plan to further explore other algorithms as well, such as the algorithm provided by Zhao et al [21].

Second, due to lack of space, for each feature-selection category (i.e., filter method, wrapper method, and embedded method), we chose to explore only one algorithm. There are many other algorithms that could be used in each category.

Third, there are inherit limitations for each feature-selection approach [4]. For instance, filter methods tend to select re-

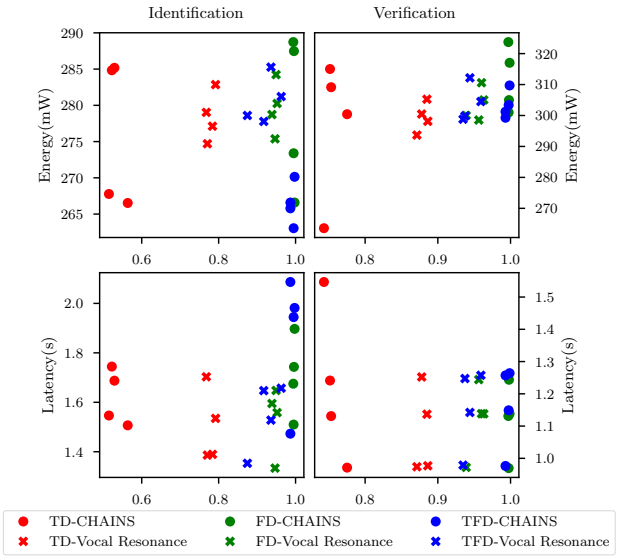


Figure 3: Accuracy, energy use and latency for the selected feature sets on both datasets and feature sets. The x-axis represents the accuracy and the y-axes represent the energy use (mW) or latency (second). Each marker represents one feature-selection algorithm (best viewed in color).

dundant features because they measure each feature independently, and wrapper methods are computationally intensive because they train a new predictive model for each feature subset explored.

4. RELATED WORK

In this paper, we proposed speaker-authentication algorithms for wearables that have limited hardware. Therefore, we describe two categories of related work: speaker-authentication algorithms, and resource efficient algorithms for wearables.

Speaker-authentication algorithms: Reynolds et al. extracted MFCCs from FD and used a GMM to train user-specific models for speaker authentication [18]. They derived a Universal Background Model (UBM) to model general, person-independent feature characteristics. By computing and comparing the probabilities of incoming voices to each user-specific model, the system identifies the user or verifies the user. There are promising efforts using deep learning for speaker authentication, which provides high accuracy [20]. Nevertheless, deep-learning algorithms are computationally complex and it is non-trivial to port them into wearables [1].

Resource-efficient algorithms: Rawassizadeh et al. created an energy-efficient frequent itemset mining algorithm [14] that can run on a smartwatch [16]. A recent work proposed to use a resource-efficient natural-language processing algorithm on a smartwatch [13]. Ravi et al. proposed a deep-learning approach on low-power wearable devices for human activity recognition [12]. This system used both time-domain and frequency-domain features extracted from inertial sensor data. Borazio et al. proposed a human-activity recognition system using time-domain features on wearable devices along with the user's survey data [3]. It used a Support Vector Machine (SVM) to recognize the activity. These works improve resource efficiency by optimizing algorithms; in this paper, however, we focus on trimming the input.

5. CONCLUSION

In this paper, we examine the challenge of computing speaker recognition on wearable platforms. Specifically, we discussed reducing resource use (energy use, response time) by trimming the input through careful feature selections, while maintaining accuracy. For our experiments, we analyzed four different feature-selection algorithms, including PC, SFS, RFE and PCA on two datasets, i.e., CHAINS and Vocal Resonance. We used three different feature sets, including time-domain features, frequency-domain features, and the combination of both, for speaker identification and verification. We evaluated accuracy metrics, energy use and latency on different datasets and features selected by different algorithms. Our results show that Principal Component Analysis (PCA) with frequency-domain features had the highest accuracy, Pearson Correlation (PC) with time-domain features had the lowest energy use, and recursive feature elimination (RFE) with frequency-domain features had the least latency. Speaker-authentication algorithms for wearable devices could choose feature sets and feature-selection algorithm based on their needs for accuracy, energy use, or latency.

6. ACKNOWLEDGMENTS

We thank Shengjie Bi, Jun Gong, and Ronald Peterson for their advice and anonymous reviewers for their helpful review comments. This research results from a research program at the Institute for Security, Technology, and Society at Dartmouth College, supported by the National Science Foundation under award number CNS-1329686. The views and conclusions contained in this document are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of the sponsors.

7. REFERENCES

[1] Sourav Bhattacharya and Nicholas D. Lane. Sparsification and separation of deep learning layers for constrained resource inference on wearables. In *Proceedings of the ACM Conference on Embedded Network Sensor Systems (SenSys)*, pages 176–189. ACM, 2016. DOI 10.1145/2994551.2994564.

[2] Frédéric Bimbot, Jean-Franc Bonastre, Corinne Fredouille, Guillaume Gravier, Ivan Magrin-Chagnolleau, Sylvain Meignier, Téva Merlin, Javier Ortega-García, Dijana Petrovska-Delacrétaz, and Douglas A. Reynolds. A tutorial on text-independent speaker verification. *EURASIP Journal on Advances in Signal Processing*, 2004(4):430–451, 2004. DOI 10.1155/s1110865704310024.

[3] Marko Borazio and Kristof Van Laerhoven. Using time use with mobile sensor data: a road to practical mobile activity recognition? In *Proceedings of the International Conference on Mobile and Ubiquitous Multimedia*, page 20. ACM, 2013. DOI 10.1145/2541831.2541850.

[4] Girish Chandrashekar and Ferat Sahin. A survey on feature selection methods. *Computers and Electrical Engineering*, 40(1):16–28, January 2014. DOI 10.1016/j.compeleceng.2013.11.024.

[5] Cory Cornelius, Zachary Marois, Jacob Sorber, Ron Peterson, Shrirang Mare, and David Kotz. Vocal resonance as a biometric for pervasive wearable devices. Technical Report TR2014-747, Dartmouth Computer Science, February 2014. Online at http://www.cs.dartmouth.edu/reports/TR2014-747.pdf.

[6] Fred Cummins, Marco Grimaldi, Thomas Leonard, and Juraj Simko. The CHAINS corpus: Characterizing individual speakers. In *Proceedings of Speech and Computer (SPECOM)*, volume 6, pages 431–435, 2006. Online at http://chains.ucd.ie/docs/chains_corpus_specom2006.pdf.

[7] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society, Series B (Methodological)*, 39(1):1–38, 1977. DOI 10.2307/2984875.

[8] Hua Huang and Shan Lin. Toothbrushing monitoring using wrist watch. In *Proceedings of the ACM Conference on Embedded Network Sensor Systems (SenSys)*, pages 202–215. ACM, November 2016. DOI 10.1145/2994551.2994563.

[9] Tomi Kinnunen and Haizhou Li. An overview of text-independent speaker recognition: From features to supervectors. *Speech Communication*, 52(1):12–40, January 2010. DOI 10.1016/j.specom.2009.08.009.

[10] Ron Kohavi and George H. John. Wrappers for feature subset selection. *Artificial Intelligence*, 97(1-2):273–324, December 1997. DOI 10.1016/s0004-3702(97)00043-x.

[11] Hong Lu, A. J. Bernheim Brush, Bodhi Priyantha, Amy K. Karlson, and Jie Liu. Speakersense: Energy efficient unobtrusive speaker identification on mobile phones. In Kent Lyons, Jeffrey Hightower, and Elaine M. Huang, editors, *Proceedings of the International Conference on Pervasive Computing*, volume 6696, pages 188–205. Springer, June 2011. DOI 10.1007/978-3-642-21726-5_12.

[12] Daniele Ravi, Charence Wong, Benny Lo, and Guang-Zhong Yang. A deep learning approach to on-node sensor data analytics for mobile or wearable devices. *IEEE Journal of Biomedical and Health Informatics*, 21(1):56–64, January 2017. DOI 10.1109/jbhi.2016.2633287.

[13] Reza Rawassizadeh, Chelsea Dobbins, Manouchehr Nourizadeh, Zahra Ghamchili, and Michael Pazzani. A natural language query interface for searching personal information on smartwatches. In *IEEE International Conference on Pervasive Computing, WristSense workshop (Percom '17)*, 2017. Online at https://arxiv.org/pdf/1611.07139.

[14] Reza Rawassizadeh, Elaheh Momeni, Chelsea Dobbins, Joobin Gharibshah, and Michael Pazzani. Scalable daily human behavioral pattern mining from multivariate temporal data. *IEEE Transactions on Knowledge and Data Engineering*, 28(11):3098–3112, November 2016. DOI 10.1109/tkde.2016.2592527.

[15] Reza Rawassizadeh, Blaine A. Price, and Marian Petre. Wearables: Has the age of smartwatches finally arrived? *Communications of the ACM*, 58(1):45–47, December 2015. DOI 10.1145/2629633.

[16] Reza Rawassizadeh, Martin Tomitsch, Manouchehr Nourizadeh, Elaheh Momeni, Aaron Peery, Liudmila Ulanova, and Michael Pazzani. Energy-efficient integration of continuous context sensing and prediction into smartwatches. *Sensors*, 15(9):22616–22645, September 2015. DOI 10.3390/s150922616.

[17] Douglas A. Reynolds. Speaker identification and verification using Gaussian mixture speaker models. *Speech Communication*, 17(1-2):91–108, August 1995. DOI 10.1016/0167-6393(95)00009-d.

[18] Douglas A. Reynolds, Thomas F. Quatieri, and Robert B. Dunn. Speaker verification using adapted Gaussian mixture models. *Digital Signal Processing*, 10(1-3):19–41, January 2000. DOI 10.1006/dspr.1999.0361.

[19] Douglas A. Reynolds and Richard C. Rose. Robust text-independent speaker identification using gaussian mixture speaker models. *IEEE Transactions on Speech and Audio Processing*, 3(1):72–83, January 1995. DOI 10.1109/89.365379.

[20] E. Variani, X. Lei, E. McDermott, I. L. Moreno, and J. Gonzalez-Dominguez. Deep neural networks for small footprint text-dependent speaker verification. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4052–4056, 2014. DOI 10.1109/ICASSP.2014.6854363.

[21] Xiaojia Zhao, Yuxuan Wang, and DeLiang Wang. Robust Speaker Identification in Noisy and Reverberant Conditions. *IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP)*, 22(4):836–845, 2014. DOI 10.1109/TASLP.2014.2308398.