

Personalizing Keyword Spotting with Speaker Information

Beltrán Labrador
Google
Zurich, Switzerland
blabrador@google.com

Pai Zhu
Google
New York, USA
paizhu@google.com

Guanlong Zhao
Google
New York, USA
guanlongzhao@google.com

Angelo Scorza Scarpati
Google
New York, USA
angelos@google.com

Quan Wang
Google
New York, USA
quanw@google.com

Alicia Lozano-Diez
Audias, Escuela Politécnica Superior
Universidad Autónoma de Madrid
Madrid, Spain
alicia.lozano@uam.es

Ignacio Lopez-Moreno
Google
New York, USA
elnota@google.com

Abstract—Keyword spotting systems often struggle to generalize to a diverse population with various accents and age groups. To address this challenge, we propose a novel approach that integrates speaker information into keyword spotting using Feature-wise Linear Modulation (FiLM), a recent method that allows models to learn from different data inputs and features. We explore both Text-Dependent and Text-Independent speaker recognition systems to extract speaker information, and we experiment on extracting this information from both the input audio and pre-enrolled user audio. Evaluating our systems on a diverse dataset, our primary approach yields a notable 2.6% relative improvement on Equal Error Rate overall, particularly improving performance by 5.9% for children under 12 years old and up to 24% for underrepresented speaker groups. Moreover, our proposed approach only requires a small 1% increase in the number of parameters, with a minimum impact on latency and computational cost, which makes it a practical solution for real-world applications.

Index Terms—Keyword spotting, target speaker personalization, AI fairness

I. INTRODUCTION

Keyword spotting (KWS) is the task of detecting specific words or phrases from an audio stream. This technology is widely used in different applications such as speech data mining, audio indexing, and wake-up word detection, which is often used to initiate an interaction with voice assistants in mobile phones, smart speakers, and other internet of things (IoT) devices (e.g. “Okay Google”, “Hey Siri”, or “Alexa”) [1]–[4].

A robust KWS system is expected to generalize to queries that may contain diverse accents, different age groups, and varying acoustic environments. A straightforward solution is to increase the KWS model capacity and train it with a diverse dataset. However, in many use cases, keyword spotting

applications are intended for low-resource devices that are characterized for having memory, computation and power constraints. This requires KWS systems to be lightweight with a relatively small memory and power footprint [5]–[13]. Therefore, parameter-efficient modeling strategies are favored over simply increasing the model size.

Many voice products provide interfaces to allow users to enroll their voice identity [14] to customize the user experience. The enrollment process generally takes place in the form of computing speaker embedding vectors from user-provided enrollment utterances. A speaker recognition model can be used to compute these speaker embeddings such as i-vectors, x-vectors, or d-vectors [15]–[17].

Using auxiliary speaker information is a common approach to adapt neural systems to different speaker characteristics. For instance, in the Automatic Speech Recognition (ASR) task, feature space maximum likelihood linear regression (fMLLR) features have been shown to outperform traditional features (e.g. MFCC) thanks to the speaker adaptation process [18]. ASR systems have also been directly conditioned with speaker embeddings [19]. These speaker embeddings can also be used to improve the performance of Voice Activity Detection (VAD) systems [20], [21]. These so-called “Personal VAD” systems typically use a small neural network to predict whether a given frame of audio contains speech, and if the speech is produced by an enrolled speaker or a non-enrolled speaker. The speaker embedding information is used to condition the neural network on the target speaker’s identity, which can help the network to learn more discriminative features for the target speaker’s voice on a computational constricted scenario.

For the keyword spotting task there has been some previous efforts on learning to customize the keyword detection to a target speaker. In [22], the authors use a speech enhancement frontend based on VoiceFilter [23] to perform speech extraction in order to only feed the target speaker audio to the keyword spotting system. Furthermore, in [24], by leveraging speaker information, multi-task learning has been applied to jointly train a system to discriminate between keywords and speakers, and afterwards perform a task specific adaptation. However, this multi-task approach may suffer from

The authors thank Joaquín González Rodríguez, Alex Park, Doroteo Torre Toledano, Jacob Bartel, Andre Perunicic, Daniel Ramos, Françoise Beaufays, and Pedro Moreno Mengibar for their help. Beltrán Labrador performed this work as an intern at Google and as a PhD student at the Universidad Autónoma de Madrid and was partially supported by FPI RTI2018-098091-B-I00, MCIU/AEI/10.13039/501100011033/FEDER, UE and PID2021-125943OB-I00, MCIN/AEI/10.13039/501100011033/FEDER, UE from the Spanish Ministerio de Ciencia e Innovación, Agencia y del Fondo Europeo de Desarrollo Regional.

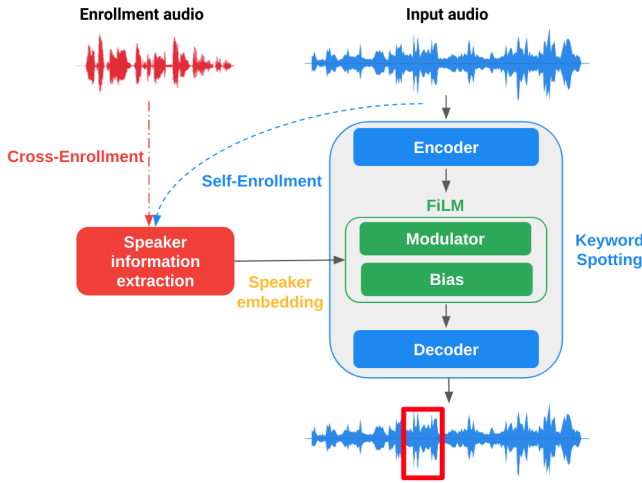


Fig. 1. We personalize the keyword detection with target speaker information applying FiLM over the encoder output logits with a speaker embedding. In the Self-Enrollment approach, this embedding vector is extracted directly from the input audio, while in the Cross-Enrollment strategy it is extracted from an enrollment audio spoken by the same individual.

the inefficiency as a result of trying to address two tasks simultaneously, as well as reduced model interpretability and increased computational complexity [25].

This work is inspired by the findings in [26], where the authors show that by conditioning a small KWS system on different locale indices they were able to improve the system’s performance in multilingual contexts. Thus, we propose to use speaker embeddings extracted from a pretrained speaker recognition system to condition a KWS system using Feature-wise Linear Modulation (FiLM) [27], an effective method to integrate and understand multiple sources of information. This allows the model to adapt to different speaking styles while keeping the restrictive computational and memory constraints.

In our experiments, we compare the performance when conditioning the KWS detection using Text-Dependent (TD) speaker embeddings with respect to employing Text-Independent (TI) representations. We analyze the results on a diverse dataset, considering various locales and age groups. Additionally, we introduce a robust training strategy to improve the system’s suitability for production scenarios.

The rest of this paper is organized as follows. Section II introduces the baseline KWS system, the speaker recognition models, as well as how we personalize keyword detection. Section III details the experimental setup including data preparation, system descriptions, and training configurations. Section IV presents the results for different models and population groups. Finally, we conclude the paper in Section V.

II. METHODS

A. Baseline

Our baseline is an end-to-end neural KWS system optimized for low-resource use cases, as described in [3]. This model has an encoder-decoder architecture with a total of 350K parameters, where the encoder has 4 Singular Value Decomposition Filter (SVDF) layers [28] with 576 nodes in each layer and a memory of 6 frames. Each SVDF layer is

followed by a bottleneck layer of size 64. The decoder has three SVDF layers with 32 nodes each and a memory of 32 frames. This system f is trained with examples composed of feature sequences (denoted as X) and corresponding label sequences (denoted as Y) that identify the keyword frames. We minimize the cross-entropy (CE) loss by finding optimal trainable parameters θ_{base} for the keyword detection task. The training dataset is mixed with all speakers S .

$$\theta_{base} = \underset{\theta}{\operatorname{argmin}} \mathbb{E}(x, y) [L_{CE}(f(x; \theta), y)] \quad (1)$$

where $(x, y) \in \bigcup_{s=1..S} (X_s, Y_s)$

we use $\mathbb{E}(\ast)[\dots]$ to denote expectation over \ast . More details of this baseline can be found in [3].

B. Speaker Embedding

We employ two distinct speaker verification systems to extract speaker embeddings. The first of these systems is a Text-Dependent (TD) system [17], which imposes the requirement that the speaker utters a specific phrase, in our case, “Okay/Hey Google”. We also experiment with a Text-Independent (TI) system, which operates without any speech constraint, allowing to perform speaker verification on spontaneous, free speech.

The TD system [17] has a compact memory footprint of 235k parameters. To train this system, we extract speech segments containing only the target keyword, a task facilitated by a pretrained keyword spotting system. This TD system consists of three Long Short-Term Memory (LSTM) layers with projection, each with 128 memory cells. A linear transformation layer follows the final projected LSTM layer to produce a 64-dimensional speaker embedding.

For the Text-Independent speaker encoder, we employ a conformer system based on the one described in [29] (Section 2.3.1), consisting of 12 conformer [30] encoder layers each of 256 dimensions, followed by an attentive temporal pooling mechanism [31], with a total of 22 million parameters, trained with the generalized end-to-end extended-set softmax (GE2E-XS) loss [17], [32]. We use this system to extract a 256-dimensional speaker embedding.

C. Personalizing detection with speaker embedding

To condition the keyword detection with speaker information, we incorporate Feature-wise Linear Modulation (FiLM) [27] into our framework as shown in Fig. 1. Using FiLM we can apply affine transformations to intermediate layer outputs within the neural network. These transformations serve as a means of dynamically adjusting the intermediate layer outputs, allowing the model to adapt its representations based on the unique characteristics of each speaker, as captured by the speaker embedding.

$$\operatorname{FiLM}(l, \gamma, \beta) = \gamma \odot l + \beta \quad (2)$$

where \odot denotes element-wise multiplication.

The FiLM mechanism effectively learns scaling and bias functions (γ and β in equation 2), which are integrated as trainable projection layers connecting the speaker embedding. We apply this modulation to the encoding logits (l in equation 2): the layer output between the encoder and decoder networks. This allows the model to perform the keyword detection task with awareness of speaker characteristics.

$$\theta_{cond} = \underset{\theta}{\operatorname{argmin}} \mathbb{E}(x, y, s)[L_{CE}(f(x, s; \theta), y)] \quad (3)$$

where $(x, y) \in (X_s, Y_s), s \in S$

We update the parameters θ_{cond} by minimizing the cross-entropy loss, given feature sequences X_s , label sequences Y_s and conditioned by the enrollment speaker embeddings stored in S .

III. EXPERIMENTAL SETUP

In all experiments, we use the same train and evaluation datasets, front-end features, and data augmentations. When handling user data, we abide by Google’s AI principles [33] and privacy principles [34].

A. Data description and preparation

The training and evaluation datasets consist of vendor-provided data. No user data were used in these experiments. Text prompts were provided to vendors, who recorded their spoken utterances based on the given transcripts. The subset that contains the targeted keyword (“Okay/Hey Google”) are referred as the positive dataset and otherwise the negative dataset. We divided the collected datasets into training, development and evaluation datasets, with no overlapping speakers. The training set has a total of 2600h of positive utterances and 2400h of negative utterance, the validation set contains 500h of positive and 400h of negative utterances, and the test set includes 670h of positive and 520h of negative data. This test set includes subsets divided by various English locales/accents and age groups (all ages and under 12 years old), with each subset having a minimum of 9 hours of data.

Moreover, the training set has been augmented with different transformations using room impulse simulations and varying degrees of noise and reverberation, producing 25 augmented copies for each original utterance [35].

Our dataset comprises diverse English accents (including those from the US, India, UK, and Australia) and a spectrum of acoustic conditions (e.g., recordings from inside vehicles and various background noise scenarios). It exhibits a balanced gender distribution and it is meticulously designed to maintain equal proportions of near-field and far-field audio recordings.

For these experiments, we extend the datasets by introducing enrollment data, where each utterance is augmented by pairing it with a corresponding same-speaker positive enrollment utterance. As we construct these datasets, our aim is to simulate the conditions of a production environment where users typically enroll their voices during their initial set up with the device, resulting in the extraction of speaker embeddings from enrollment utterances.

B. System description

We evaluate the performance of three different approaches, depending on the speaker embedding types and how we simulate the enrollment utterances:

- **Text-Independent Self-Enrollment:** In this setup, we extract a Text-Independent speaker embedding from the same utterance used as input for the keyword detection. This implies that the speaker information that conditions the KWS encoder’s output, is extracted from the exact same speech sample in which we aim to identify the target keyword.
- **Text-Independent Cross-Enrollment:** Here, the speaker embedding is extracted from a simulated enrollment utterance (by choosing a random utterance from the same speaker). This setup is closer to a production environment, where the enrollment utterances in production are pre-enrolled and are different from the query utterance.
- **Text-Dependent Cross-Enrollment:** We use a Text-Dependent (TD) speaker embedding from the simulated enrollment utterance mentioned above. To obtain the TD speaker embedding, we first use a pretrained keyword spotting model to extract the segment of the enrollment utterance where the keyword is spoken. Therefore, in this approach, the utterance segment that is used to compute the speaker embedding is constrained to the keyword, thus containing less variability and noise.

Note that the Text-Dependent Self-Enrollment scenario is not viable in our setup. This limitation comes from the absence of keyword in negative utterances, and the restriction of Text-Dependent SID systems to extract the speaker embedding from the target keyword segment.

C. Training a robust model for non-enrollment conditions

In a real-world scenario, enrollment utterances may not always be available due to a failed or skipped enrollment stage. It is essential that the model can still accurately detect the keyword without relying on the speaker’s enrollments. To ensure model’s robustness, we have trained a model mixing utterances containing the speaker embedding with utterances without any speaker embedding. Specifically, we randomly replace the enrollment utterance speaker embedding of dimension D with a same dimension vector of constant value $1/D$. This forces the model to learn to detect the keyword both with and without the speaker embedding information.

IV. RESULTS AND DISCUSSION

In this section, we present the results of the keyword detection task, focusing on the Equal Error Rate (EER) as single value metric and Detection Error Tradeoff (DET) curves showcasing model performance at various operating points.

Table I and Fig. 2 summarize the main results of these experiments. Specifically, we evaluate the three variations mentioned before: Text-Independent Self-Enrollment, Text-Independent Cross-Enrollment, and Text-Dependent systems. For each of these systems, we report the EER and DET curves across the entire dataset, as well as stratified by different

TABLE I
PERFORMANCE IN EER (%) OF THE SYSTEMS ON DIFFERENT POPULATION GROUPS (ENGLISH LOCALES/ACCENTS & AGE)

EER (%)	All locales		India		United States		Great Britain		Australia	
	All ages	<12	All ages	<12	All age	<12	All ages	<12	All ages	<12
Baseline	1.93%	3.59%	3.31%	3.7%	3.74%	4.98%	0.6%	1.85%	0.78%	1.07%
Text-Indep. Self-Enrollment	1.57%	3.01%	2.52%	3.2%	3.41%	3.96%	0.51%	1.66%	0.73%	0.78%
Text-Indep. Cross-Enrollment	2.12%	3.75%	3.81%	2.88%	3.85%	5.18%	0.6%	1.7%	0.71%	0.99%
Text-Dep.	1.88%	3.38%	3.26%	2.79%	3.75%	4.43%	0.63%	1.59%	0.67%	0.84%
Relative improvement (%)										
Text-Indep. Self-Enrollment vs Baseline	-18.7%	-16.2%	-23.9%	-13.5%	-8.8%	-20.5%	-15.0%	-10.3%	-6.4%	-27.1%
Text-Indep. Cross-Enrollment vs Baseline	9.8%	4.5%	15.1%	-22.2%	2.9%	4.0%	0.0%	-8.1%	-9.0%	-7.5%
Text-Dep. vs Baseline	-2.6%	-5.9%	-1.5%	-24.6%	0.3%	-11.0%	5.0%	-14.1%	-14.1%	-21.5%

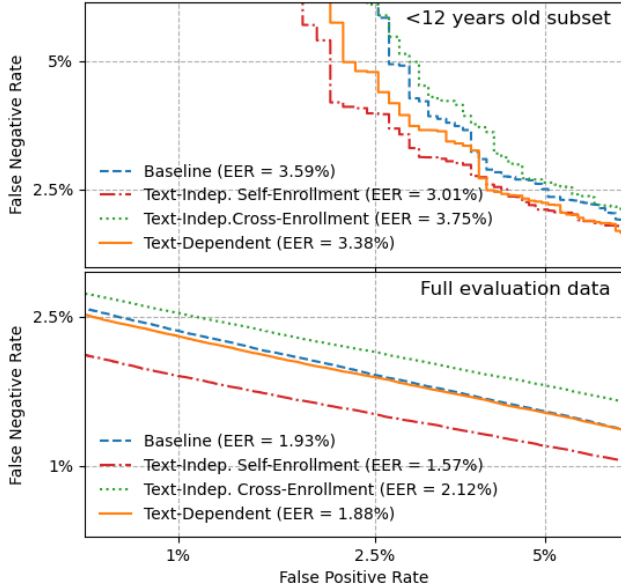


Fig. 2. Detection Error Trade-off (DET) curves illustrating the performance of the different approaches on the full evaluation data and on the under 12 years old evaluation subset.

locales and age groups (all ages and those under 12 years old). Furthermore, we show the relative improvement over the baseline for each configuration.

We observe that the Text-Independent Self-Enrollment approach yielded the most substantial improvement over the baseline model, achieving an 18% relative improvement in terms of EER. This underscores the potential of the proposed strategy, as it leverages speaker embedding extracted from the same target utterance. However, it is crucial to note that this approach leads to a $\sim 11\times$ increase in computational demands and system latency at inference time, due to the substantial increase of operations within the combined network. Therefore, it may be less practical for production environments, specially those with strict real-time processing requirements.

Conversely, in the Text-Independent Cross-Enrollment scenario, where speaker embeddings are derived from enrollment utterances, we generally observed a degradation in performance. This result can likely be attributed to the high dimensionality of the 256-dimensional embedding, which may underfit the single layer FiLM. Furthermore, this embedding is built on top of diverse spoken content, which seems less effective at capturing speaker characteristics useful for this scenario as compared to extracting an embedding from just the target keyword segment, as in the Text-Dependent approach.

Therefore, in the Text-Dependent scenario, where a 64-dimensional speaker embedding is extracted from the enrollment utterance keyword segment, is expected to have a more reliable representation of the speaker characteristics. This approach has a notable 2.6% relative improvement over the baseline, a 5.9% EER relative improvement in children data, and as far as a remarkable 24% enhancement when dealing with children data in the distinctly accented India locale. Importantly, the system barely adds any additional computation or latency in the production environment, as the speaker embeddings are pre-computed during the enrollment stage when the devices are first set up.

TABLE II
PERFORMANCE IN EER (%) OF THE ROBUST TRAINING FOR THE TEXT-DEP. APPROACH EXPLAINED IN SUBSECTION III-C ON DIFFERENT SPEAKER ENROLLMENT CONDITIONS

	With spk. embedding	Without spk. embedding
Baseline	-	1.93 %
Text-Dep.	1.88 %	39.54 %
Robust Text-Dep.	1.85 %	2.03 %

Finally, we compare enrollment and non enrollment scenarios (with no pre-enrolled utterance) to assess the impact of enrollment utterances on the keyword spotting performance and underscore the importance of a robust model. As shown in Table II, in the absence of enrollment data, the TD-conditioned keyword spotting system experiences a complete failure. However, with our robust training approach (Section III-C), we ensure that the system remains consistent with the baseline performance. Furthermore, when provided with an enrollment speaker embedding, it yields a remarkable 4.1% EER relative improvement over the baseline, enhancing also the regular TD approach by introducing a regularization effect, showing the adaptability of our approach, particularly in situations where speaker enrollment data is not present.

V. CONCLUSION

In this paper, we propose a novel method to personalize keyword spotting by integrating speaker information through FiLM modulation. We compare various approaches and speaker embedding types to identify the most effective strategies, demonstrating notable performance improvement, specifically when handling non-standard speech patterns and diverse speaker profiles. This approach holds the potential to benefit a wide range of applications, by ensuring that the technology can effectively understand and respond to the unique voices and needs of diverse individuals, making the technology more accessible, adaptive, and inclusive.

REFERENCES

- [1] Ivan Lopez Espejo, Zheng-Hua Tan, John Hansen, and Jesper Jensen, "Deep spoken keyword spotting: An overview," *IEEE Access*, vol. 10, pp. 4169–4199, Jan. 2022.
- [2] Siddharth Sigtia, John Bridle, Hywel Richards, Pascal Clark, Erik Marchi, and Vineet Garg, "Progressive voice trigger detection: Accuracy vs latency," in *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2021, pp. 6843–6847.
- [3] Raziel Alvarez and Hyun-Jin Park, "End-to-end streaming keyword spotting," in *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019, pp. 6336–6340.
- [4] Bowen Shi, Ming Sun, Chieh-Chi Kao, Viktor Rozgic, Spyros Matsoukas, and Chao Wang, "Compression of Acoustic Event Detection Models with Quantized Distillation," in *Proc. Interspeech 2019*, 2019, pp. 3639–3643.
- [5] Ming Sun, David Snyder, Yixin Gao, Varun Nagaraja, Mike Rodehorst, Sankaran Panchapagesan, Nikko Strom, Spyros Matsoukas, and Shiv Vitaladevuni, "Compressed Time Delay Neural Network for Small-Footprint Keyword Spotting," in *Proc. Interspeech 2017*, 2017, pp. 3607–3611.
- [6] Kenichi Kumatani, Sankaran Panchapagesan, Minhua Wu, Minjae Kim, Nikko Strom, Gautam Tiwari, and Arindam Mandai, "Direct modeling of raw audio with DNNs for wake word detection," in *Proc. ASRU*, 2017, pp. 252–257.
- [7] Guoguo Chen, Carolina Parada, and Georg Heigold, "Small-footprint keyword spotting using deep neural networks," in *ICASSP 2014 - 2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2014, pp. 4087–4091.
- [8] Tara N. Sainath and Carolina Parada, "Convolutional neural networks for small-footprint keyword spotting," in *Proc. Interspeech 2015*, 2015, pp. 1478–1482.
- [9] Ming Sun, Anirudh Raju, George Tucker, Sankaran Panchapagesan, Gengshen Fu, Arindam Mandal, Spyros Matsoukas, Nikko Strom, and Shiv Vitaladevuni, "Max-pooling loss training of long short-term memory networks for small-footprint keyword spotting," in *2016 IEEE Spoken Language Technology Workshop (SLT)*, 2016, pp. 474–480.
- [10] Qu Yang, Qi Liu, and Haizhou Li, "Deep residual spiking neural network for keyword spotting in low-resource settings," in *Proc. Interspeech 2022*, 2022, pp. 3023–3027.
- [11] Hyun-Jin Park, Patrick Violette, and Niranjana Subrahmanya, "Learning to detect keyword parts and whole by smoothed max pooling," in *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 7899–7903.
- [12] Hyun-Jin Park, Pai Zhu, Ignacio Lopez Moreno, and Niranjana Subrahmanya, "Noisy Student-Teacher Training for Robust Keyword Spotting," in *Proc. Interspeech 2021*, 2021, pp. 331–335.
- [13] Beltrán Labrador, Guanlong Zhao, Ignacio López Moreno, Angelo Scorza Scarpati, Liam Fowl, and Quan Wang, "Exploring sequence-to-sequence transformer-transducer models for keyword spotting," in *ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2023.
- [14] Quan Wang and Ignacio Lopez Moreno, "Version control of speaker recognition systems," *arXiv preprint arXiv:2007.12069*, 2020.
- [15] Patrick Kenny, Gilles Boulianne, and Pierre Dumouchel, "Eigenvoice modeling with sparse training data," *IEEE Transactions on Speech and Audio Processing*, vol. 13, no. 3, pp. 345–354, 2005.
- [16] David Snyder, Daniel Garcia-Romero, Gregory Sell, Daniel Povey, and Sanjeev Khudanpur, "X-vectors: Robust DNN embeddings for speaker recognition," in *ICASSP 2018 - 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 5329–5333.
- [17] Li Wan, Quan Wang, Alan Papir, and Ignacio Lopez Moreno, "Generalized end-to-end loss for speaker verification," in *ICASSP 2018 - 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 4879–4883.
- [18] Sree Hari Krishnan Parthasarathi, Björn Hoffmeister, Spyros Matsoukas, Arindam Mandal, Nikko Ström, and Sri Garimella, "fMLLR based feature-space speaker adaptation of DNN acoustic models," in *Proc. Interspeech 2015*, 2015, pp. 3630–3634.
- [19] Murali Karthick Baskar, Tim Herzig, Diana Nguyen, Mireia Diez, Tim Polzehl, Lukas Burget, and Jan Černocký, "Speaker adaptation for Wav2vec2 based dysarthric ASR," in *Proc. Interspeech 2022*, 2022, pp. 3403–3407.
- [20] Shaojin Ding, Quan Wang, Shuo-Yiin Chang, Li Wan, and Ignacio Lopez Moreno, "Personal VAD: Speaker-Conditioned Voice Activity Detection," in *Proc. The Speaker and Language Recognition Workshop (Odyssey 2020)*, 2020, pp. 433–439.
- [21] Shaojin Ding, Rajeev Rikhye, Qiao Liang, Yanzhang He, Quan Wang, Arun Narayanan, Tom O'Malley, and Ian McGraw, "Personal VAD 2.0: Optimizing Personal Voice Activity Detection for On-Device Speech Recognition," in *Proc. Interspeech 2022*, 2022, pp. 3744–3748.
- [22] Qijie Shao, Jingyong Hou, Yanxin Hu, Qing Wang, Lei Xie, and Xin Lei, "Target speaker extraction for customizable query-by-example keyword spotting," in *2021 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, 2021, pp. 672–678.
- [23] Quan Wang, Hannah Muckenhirn, Kevin Wilson, Prashant Sridhar, Zelin Wu, John R. Hershey, Rif A. Saurous, Ron J. Weiss, Ye Jia, and Ignacio Lopez Moreno, "VoiceFilter: Targeted Voice Separation by Speaker-Conditioned Spectrogram Masking," in *Proc. Interspeech 2019*, 2019, pp. 2728–2732.
- [24] Seunghan Yang, Byeonggeun Kim, Inseop Chung, and Simyung Chang, "Personalized keyword spotting through multi-task learning," in *Proc. Interspeech 2022*, 2022, pp. 1881–1885.
- [25] Rich Caruana, "Multitask learning," *Machine learning*, vol. 28, pp. 41–75, 1997.
- [26] Pai Zhu, Hyun Jin Park, Alex Park, Angelo Scorza Scarpati, and Ignacio Lopez Moreno, "Locale encoding for scalable multilingual keyword spotting models," in *ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2023.
- [27] Ethan Perez, Florian Strub, Harm Vries, Vincent Dumoulin, and Aaron Courville, "FiLM: Visual reasoning with a general conditioning layer," in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2018, vol. 32, pp. 3942–3951.
- [28] Preetum Nakkiran, Raziel Alvarez, Rohit Prabhavalkar, and Carolina Parada, "Compressing deep neural networks using a rank-constrained topology," in *Proc. Interspeech 2015*, 2015, pp. 1473–1477.
- [29] Quan Wang, Yiling Huang, Han Lu, Guanlong Zhao, and Ignacio Lopez Moreno, "Highly efficient real-time streaming and fully on-device speaker diarization with multi-stage clustering," *arXiv:2210.13690*, 2022.
- [30] Anmol Gulati, James Qin, Chung-Cheng Chiu, Niki Parmar, Yu Zhang, Jiahui Yu, Wei Han, Shibo Wang, Zhengdong Zhang, Yonghui Wu, and Ruoming Pang, "Conformer: Convolution-augmented Transformer for Speech Recognition," in *Proc. Interspeech 2020*, 2020, pp. 5036–5040.
- [31] Quan Wang, Yang Yu, Jason Pelecanos, Yiling Huang, and Ignacio Lopez Moreno, "Attentive Temporal Pooling for Conformer-Based Streaming Language Identification in Long-Form Speech," in *Proc. The Speaker and Language Recognition Workshop (Odyssey 2022)*, 2022, pp. 255–262.
- [32] Jason Pelecanos, Quan Wang, and Ignacio Lopez Moreno, "Dr-Vectors: Decision Residual Networks and an Improved Loss for Speaker Recognition," in *Proc. Interspeech 2021*, 2021, pp. 4603–4607.
- [33] Google, "Artificial intelligence at Google: Our principles," <https://ai.google/principles>, Accessed: 2023-10-05.
- [34] "Google's privacy principles," <https://googleblog.blogspot.com/2010/01/googles-privacy-principles.html>, Accessed: 2023-10-05.
- [35] Chanwoo Kim, Ananya Misra, Kean Chin, Thad Hughes, Arun Narayanan, Tara Sainath, and Michiel Bacchiani, "Generation of large-scale simulated utterances in virtual rooms to train deep-neural networks for far-field speech recognition in google home," in *Proc. Interspeech 2017*, 2017, pp. 379–383.