Poster Abstract: Towards Speaker Identification on Resource-Constrained Embedded Devices

Markus Gallacher
Carlo Alberto Boano
markus.gallacher@tugraz.at
cboano@tugraz.at
Graz University of Technology
Graz, Austria

M.S. Arun Sankar
Utz Roedig
asankar@ucc.ie
u.roedig@cs.ucc.ie
University College Cork
Cork, Ireland

Willian T. Lunardi
Michael Baddeley
willian.lunardi@tii.ae
michael.baddeley@tii.ae
Technology Innovation Institute
Abu Dhabi, United Arab Emirates

ABSTRACT

Voice is a convenient and popular way to interact with our digital world. Besides translating speech to text, it is also possible to identify speakers based on their voice profile. To date, speaker identification has predominantly been limited to high-performance computational platforms owing to the intricate nature of the underlying algorithms. In this work, we demonstrate that it is possible to reduce model complexity by the required factor of ∼10, such that speaker identification can be made feasible for embedded devices with limited resources. We further describe and discuss novel use cases, such as voice-based presence detection and authentication, that become feasible on these class of devices.

ACM Reference Format:
Markus Gallacher, Carlo Alberto Boano, M.S. Arun Sankar, Utz Roedig, Willian T. Lunardi, and Michael Baddeley. 2023. Poster Abstract: Towards Speaker Identification on Resource-Constrained Embedded Devices. In *The 21st ACM Conference on Embedded Networked Sensor Systems (SenSys '23), November 12–17, 2023, Istanbul, Turkiye.* ACM, New York, NY, USA, 2 pages. https://doi.org/10.1145/3625687.3628387

1 INTRODUCTION

Voice is now a common interface through which users interact with digital infrastructure. The ease of using natural language as an interface has fueled its widespread adoption. A voice interface can be a standalone device such as a smart speaker (e.g., Amazon Echo, Google Home, and Apple HomePod) or integrated within devices such as smartphones or televisions. Typically, the architecture of a voice processing system is divided into a front-end, situated on the device, which is equipped with a microphone, speaker, and limited computing capabilities, and a back-end, located remotely in the cloud, responsible for voice processing.

In addition to the well-known automatic speech recognition task, Deep Learning (DL) can be employed to extract features from a voice signal for performing other tasks, such as Speaker Identification (SI), which is the process of distinguishing and recognizing speakers based on the unique characteristic of their voice. Many methods have been proposed for SI, with Deep Neural Network (DNN) being the most popular and successful approach [5]. Current research on SI has focused on classification accuracy. Model complexity and the resulting processing and storage requirements have been treated as secondary. So far, this has not been an issue, as analysis of speech signals has been carried out on powerful cloud back-ends. However, it has been recognized that complexity should be addressed due to several drivers. First, SI should ideally be processed on a front-end (i.e., a small embedded device) to avoid the processing of personal biometric information on a remote back-end. Second, SI should reduce complexity to conserve resources, either energy consumption in a back-end or materials to construct a front-end.

In our work, we follow the aforementioned argumentation and describe how it is possible to reduce SI algorithms in their complexity such that processing on small embedded devices becomes viable without losing the required accuracy. We start our work with `AM-MobileNet1D`, a state-of-the-art model that has a flash memory footprint of 10.5 MB, and show that it is possible to reduce its size to 433 kB for the same number of speakers without significantly losing accuracy. Thus, it becomes possible to locate the SI task on small embedded devices with less than 1 MB of flash memory, which opens a new set of application scenarios. For example, small embedded devices equipped with a simple microphone can be used to overhear conversations in a space and can identify present individuals. It may also be desirable that small embedded devices that do not provide classical user interfaces, such as a keyboard/screen, implement access control for device configuration.

The specific contributions of our work are twofold:

- *SI use cases.* We describe and discuss novel use cases for SI in the context of embedded systems. From these use cases, we derive SI performance (e.g., accuracy) and complexity (e.g., storage, computation) requirements. We show that existing solutions do not meet those requirements.
- *Embedded SI implementation.* We show how a model (i.e., `AM-MobileNet1D`) for SI can be tuned to fulfill the aforementioned requirements. We provide a performance analysis of this solution and describe it in the context of our SI use cases.

2 THE SPECTRUM OF EMBEDDED USE CASES FOR SPEAKER IDENTIFICATION

Speaker Verification (SV) and SI both utilize analogous techniques in voice analysis, yet they serve distinct purposes. Specifically, SI is predominantly performed in a supervised manner, utilizing labeled data where the identity of each speaker is known beforehand. This approach allows the system to classify an unknown voice sample into one of the known categories based on training data.

SenSys '23, November 12–17, 2023, Istanbul, Turkiye

ACM ISBN 979-8-4007-0414-7/23/11.
https://doi.org/10.1145/3625687.3628387

Table 1: Selected SI models and their details, with size values in Megabytes (MB) and million params. (M), and their performance values in accuracy (Acc.) or equal error rate (EER).

Model	MB	M	# Speakers	Performance
SincNet [4]	91.2	22.8	462	Acc.=98.85%
AM-MobileNet1D [4]	11.6	2.8	462	Acc.=99.50%
VGG-M [2]	–	67	1251	Acc.=80.9%
ResNet18 [2]	–	12	1251	Acc.=89.5%
ResNet34 [2]	–	22	1251	Acc.=93.8%
FCN [3]	14.52	–	300	EER=1.235%
LCN [3]	14.39	–	300	EER=1.228%
CNN [3]	14.58	–	300	EER=1.838%
MaxOut [3]	14.56	–	300	EER=1.239%
DSC [3]	14.51	–	300	EER=1.274%
UtterIdNet [1]	268	–	1250	EER=6.63%

On the other hand, SV operates in a binary decision-making context, determining whether a given voice sample matches a claimed identity. Contrary to SI, the training for SV often leans towards an unsupervised paradigm, where the system primarily focuses on understanding the distribution of genuine voices. By learning this distribution, it becomes adept at identifying out-of-distribution samples, which often correspond to impostor voice claims.

For our use cases, we are considering small embedded devices equipped with microphones deployed in a given space. We aim to analyze overheard speech to identify the speaker(s). Classical SV, as seen in banking applications where the caller is identified, is not a suitable option. The claim of identity is inherently unknown because the embedded device typically lacks the means to obtain this information (e.g., caller ID, face recognition, a keyboard to prompt for a username, etc.). Therefore, we are focusing on SI.

In our context, SI enables a variety of application scenarios, each with distinct performance requirements. On one end of the spectrum, SI might be used to overhear speakers in space and provide an estimate of who is present at a given point in time. For such a task, a lower speaker classification accuracy may be acceptable. Conversely, SI might be employed for authentication to ease access control for device configuration. In this scenario, high accuracy is crucial, as insufficient accuracy would compromise access control. Given that state-of-the-art SI models are too complex for resource-constrained embedded devices, they must be adapted. Reducing model complexity typically impacts accuracy; some SI use cases may not be feasible, while others can be supported.

3 PRELIMINARY RESULTS

We target the nRF5340 System-on-Chip integrated into the Thingy:53 prototyping platform. This platform includes a pulse density modulation microphone and offers 512 kB of RAM and 1 MB of flash memory. The `AM-MobileNet1D` is the smallest among the list in Tab. 1, with a file size of 11.6 MB, which corresponds to approximately 10.5 MB in flash memory. We opted to shrink the `AM-MobileNet1D` by reducing its depth in terms of the number of inverted residuals, and its width through the width multiplier argument in the `AM-MobileNet1D`. The re-trained original model achieves 97.7% accuracy, and the reduced model achieves 86.6% accuracy, only a 11.1% decrease, using the same TIMIT dataset for SI. Moreover, the model's file size is reduced to 1.44 MB (-87.6%). The memory footprint of the model can be further reduced by up to a factor of 4 using full-integer quantization, which converts the weight and activation variables from floats to integers. The flash memory stores the model's architecture information and weights. In contrast, the peak RAM consumption is determined by the tensor arena size, which is defined by the number of weights and the activation's inputs and outputs of two layers needed to compute intermediate results. We use TensorFlow Lite for micro-controllers to convert the model into a `C` flat buffer array and to calculate the RAM and flash memory usage. As shown in Tab.2, the reduced model only needs 481 kB (instead of 567 kB) of RAM and 433 kB (instead of 3.1 MB) of flash memory, thus enabling embedded SI.

Table 2: Number of speakers, accuracy, and memory usage of the `AM-MobileNet1D` with full-integer quantization.

AM-MobileNet1D	# Speakers	Accuracy	Memory	
			RAM	Flash
Original	462	97.7%	567 kB	3.1 MB
Reduced	462	86.6%	481 kB	433 kB

4 DISCUSSION AND OUTLOOK

Our preliminary results demonstrate that embedded SI is feasible. This allows us to further test various models, evaluate out-of-set accuracy for SV applications, and create a running prototype on the Thingy:53. A further aspect to evaluate is the energy consumption and runtime of the shrunk model. The nRF5340 has the same current draw when recording audio data as it does during inference, averaging 6.8 mA. Thus, a brief audio recording is just as crucial as a short inference time. The TIMIT dataset contains audio samples with an average length of $\approx$ 2.7 seconds, which makes the recording relatively expensive. We employ a window size of 200 ms that slides over an audio sample with a step size of 190 ms, further increasing the number of inferences needed per audio sample. Therefore, we also aim to evaluate the model's inference time, and how different sample lengths and window sizes affect the accuracy to create a system that uses the CPU as efficiently as possible.

ACKNOWLEDGEMENTS

This work was supported by the Science Foundation Ireland under Grant number 19/FFP/6775 and 13/RC/2077_P2, and the Technology Innovation Institute under the SPiDR project.

REFERENCES

[1] A. Hajavi and A. Etemad. 2019. A Deep Neural Network for Short-Segment Speaker Recognition. In *Proc. of the 20th Conference of the International Speech Communication Association (Interspeech).* https://doi.org/10.21437/Interspeech.2019-2240

[2] M. Jakubec et al. 2021. Speaker Recognition with ResNet and VGG Networks. In *Proc. of the 31st International Conference Radioelektronika (RADIOELEKTRONIKA).* https://doi.org/10.1109/RADIOELEKTRONIKA52220.2021.9420202

[3] S. Koppula et al. 2018. Energy-Efficient Speaker Identification with Low-Precision Networks. In *Proc. of the International Conference on Acoustics, Speech and Signal Processing (ICASSP).* https://doi.org/10.1109/ICASSP.2018.8462498

[4] C. Nunes et al. 2020. AM-MobileNet1D: A Portable Model for Speaker Recognition. In *Proc. of the International Joint Conference on Neural Networks (IJCNN).* https://doi.org/10.1109/IJCNN48605.2020.9207519

[5] S.S. Tirumala and S.R. Shahamiri. 2016. A Review on Deep Learning Approaches in Speaker Identification. In *Proc. of the 8th International Conference on Signal Processing Systems.* https://doi.org/10.1145/3015166.3015210