Received: 20 April 2021 | Revised: 3 August 2021 | Accepted: 17 September 2021

DOI: 10.1049/cdt2.12035

IET Computers & Digital Techniques

The Institution of Engineering and Technology WILEY

ORIGINAL RESEARCH PAPER

Who is wearing me? TinyDL-based user recognition in constrained personal devices

Ramon Sanchez-Iborra[1] | **Antonio Skarmeta**[2]

[1]Department of Engineering and Applied Techniques, University Center of Defense, Madrid, Spain

[2]Department of Information Engineering and Communications, University of Murcia, Murcia, Spain

Correspondence

Ramon Sanchez-Iborra, Department of Engineering and Applied Techniques, University Center of Defense, San Javier Air Force Base, Madrid, Spain.
Email: ramon.sanchez@cud.upct.es

Funding information

Spanish Ministry of Science, Innovation and Universities, Grant/Award Number: ONOFRE 3 (PID2020-112675RB-C44); European Commission, Grant/Award Numbers: CyberSec4Europe (830929), INSPIRE-5Gplus (871808); Spanish Ministry for the Ecological Transition and the Demographic Challenge, Grant/Award Number: MECANO (PGE-MOVES-SING-2019-000104); Ministerio de Ciencia, Innovación y Universidades

Abstract

Deep learning (DL) techniques have been extensively studied to improve their precision and scalability in a vast range of applications. Recently, a new milestone has been reached driven by the emergence of the TinyDL paradigm, which enables adaptation of complex DL models generated by well-known libraries to the restrictions of constrained microcontroller-based devices. In this work, a comprehensive discussion is provided regarding this novel ecosystem, by identifying the benefits that it will bring to the wearable industry and analysing different TinyDL initiatives promoted by tech giants. The specific use case of automatic user recognition from data captured by a wearable device is also presented. The whole development process by which different DL configurations have been embedded in a real microcontroller unit is described. The attained results in terms of accuracy and resource usage confirm the validity of the proposal, which allows precise predictions in a highly constrained platform with limited input information. Therefore, this work provides insights into the viability of the integration of TinyDL models within wearables, which may be valuable for researchers, practitioners, and makers related to this industry.

KEYWORDS

deep learning, IoT, TinyDL, wearables

1 | INTRODUCTION

Artificial intelligence (AI) and, in particular, deep learning (DL) has received the attention from both academia and industry during the last few years given their potential to generate valuable decision support systems for a plethora of applications and processes. The prominent advances in processing hardware and computing schemes have fuelled the development of enhanced DL mechanisms that are able to deal with complex problems in a range of sectors, for example, industrial, healthcare, or daily common activities, among others [1]. Most of these processing tasks are performed in the cloud with data captured in the field, which frequently implies their wireless transmission to the Internet. This is not always the best solution considering cybersecurity and privacy issues as well as energetic aspects because many of the deployed end devices are powered by batteries.

This is the point in which TinyDL comes into play. This groundbreaking paradigm proposes the integration of DL algorithms within scarce-resource devices, for example, microcontroller units (MCUs) as defined in [2], by transforming the models produced by widely adopted libraries such as TensorFlow, PyTorch etc. To this end, the selected model is configured, trained, and generated in a non-constrained platform and, thereafter, its code is ported to be runnable in the target resource-limited unit (Figure 1). This enables the evolution of the classic Internet of Things (IoT) towards the innovative Internet of Smart Things by exploiting a rich variety of models designed with the aforementioned or similar frameworks and, therefore, not being limited to non-generalizable DL ad-hoc implementations.

The contributions of this work are the following: First, we provide a comprehensive discussion regarding the incipient TinyDL ecosystem, identifying the main benefits that it will

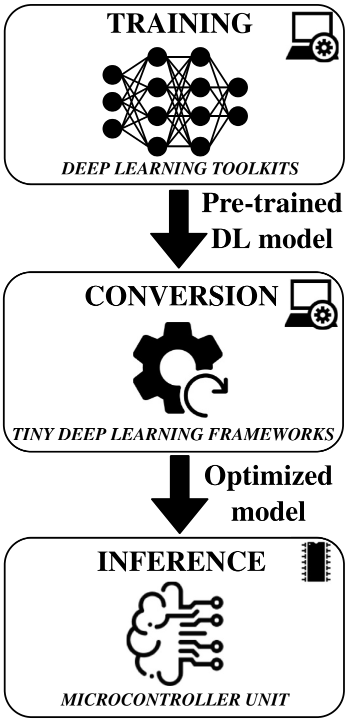

FIGURE 1 Deep learning model conversion process

bring to the wearable industry. Then, some TinyDL frameworks publicly released by big technological companies are also surveyed, and their main characteristics and limitations are analysed. Finally, as the final contribution of this work, a real use case is presented in which we integrate different multi-layer neural network (NN) configurations within a well-known MCU for the task of user recognition by just employing data captured by a wearable inertial sensing module. This issue is of relevant interest for diverse types of wearable devices in order to provide contextual services and a personalised experience. Therefore, the purpose of this work is to provide practical details about the feasibility of integrating TinyDL-based mechanisms within wearable devices with the aim of presenting useful insights to researchers, practitioners, and makers related to the wearable industry.

The rest of this work is organised as follows: Section 2 provides an overview of the related work. Section 3 explores the application of TinyDL models for enabling a variety of services in wearables. The implementation details and validation results of our case study are shown in Section 4. Finally, Section 5 closes the work, highlighting the most important findings and drawing future research lines.

2 | BACKGROUND

The introduction of DL-based models in wearables is still in its infancy; hence very few studies in this line can be found in the related literature. However, the TinyDL paradigm can be framed within the wider Tiny Machine Learning (TinyML) ecosystem [3], which has received great attention in the past. Some notable advances have also been achieved with other types of constrained IoT devices [4]. The democratization of these kind of AI-based solutions is highly relevant in order to enable the integration of intelligent mechanisms within the IoT edge as discussed in Ref. [5].

From a general perspective, that is, not focussed on a specific application, the authors of Ref. [6] exploited the capabilities of neural architecture search in order to design accurate ML models that meet the stringent memory, latency, and energy constraints of MCUs. A family of models, the so-called MicroNets, was presented and optimised for different MCUs, obtaining good performance in a variety of tasks. In Ref. [7], an incremental algorithm to enable on-device learning and not only the inference was proposed. This mechanism, based on transfer learning and k-nearest neighbour, enables devices to learn in an incremental way directly from the field; thus, it is independent from a ML training architecture. Given the low-cost of IoT units, the authors of Ref. [8] explored the benefits of deploying AI models in resource-limited devices to enable the digitalization of remote rural areas. To this end, a framework that enables such integration was presented with the aim of being exploited in small farms.

On-device visual and audio processing is a recurrent field of study. The work in Ref. [9] proposed MCUNet, a framework that includes a series of efficient libraries to jointly design the NN architecture and the inference engine. The validation experiments focussed on different tasks, namely, large-scale image recognition, object detection, and wake words inference. The attained results showed improved performance in terms of accuracy and latency in comparison with well-known libraries such as TensorFlow Lite or Cortex Microcontroller Software Interface Standard-NN (CMSIS-NN; see Section 3.1). In turn, the authors of Ref. [10] proposed the concept of attention condensers in order to obtain highly efficient DL models with limited computational requirements, targetting the task of on-device speech recognition. The presented results showed that the developed models achieve notable reduction in architectural, computational, and storage complexities in comparison to similar state-of-the-art proposals. Object recognition is another hot topic of study given its wide applicability. The work in Ref. [11] presented a battery-free camera for face recognition. The authors embedded a TinyML-based model for this task in a resource-limited unit with a photovoltaic panel for energy harvesting, which is a highly interesting technology for wearable devices [12]. This innovative unit showed an accuracy of 97%, recognizing five different faces. In a similar line, the solution proposed in Ref. [13] enables constrained devices to detect the use of face masks. The authors made use of adapted convolutional NNs (CNNs) to achieve an accuracy over 99% while keeping the memory footprint very low.

Other publications have proposed diverse types of applications by exploiting the processing flexibility of TinyML. In Ref. [14], an alcohol level detector was built using the TensorFlow Lite micro library. The implemented model enables calibration of the sensor response locally with a corrective procedure considering a variety of thermal and humidity situations, which is an interesting feature that shows the adaptation capability of these kinds of mechanisms. The authors of Ref. [15] presented a TinyML-based navigation system for autonomous mini-vehicles. This solution employs reduced CNNs for the image classification task, obtaining very low computing latencies and meeting real-time requirements. Moreover, the accuracy of the solution also improves that obtained by other recent proposals evaluated in the work. Finally, the work in Ref. [16] presented a solution for weather forecast at the edge, that is, without the support of cloud servers. The authors developed an atmospheric pressure forecast model based on tiny NNs tested on a highly constrained device, obtaining a similar prediction quality than the same NN model fully deployed on the cloud.

From the previous literature review, the large range of applications enabled by the TinyML and TinyDL paradigms can be seen. Different from previous works, first, we comprehensively explore the applicability of TinyDL in the wearable ecosystem. To this end, we discuss the main services and fields of application that can leverage the advantages brought by the integration of intelligence within these tiny devices. Further, we present the most important TinyDL toolkits developed by big companies, which are publicly available for experimentation. Finally, we show the implementation process and results of a real use case. Concretely, we have tackled the task of on-device user recognition, which we consider of high interest for wearable devices. To the author's knowledge, this is the first integration of different TinyDL models into a highly constrained real wearable device.

3 | TINY DEEP LEARNING IN WEARABLES

The wearable segment market has quickly grown during the last years and forecasts predict that the penetration of these small devices will continue to rise, reaching 1 billion by 2022 [17]. So far, wearables have presented highly limited processing capabilities, which has led to the need for an almost permanent connection of these elements with the cloud or with other devices. For example, fitness trackers are usually paired via Bluetooth with a user's smartphone in order to transfer the raw captured data that are actually processed in the non-constrained device or even in cloud servers. This may be problematic for a number of reasons. First, the excessive use of communication activities leads to greater power consumption as these tasks are more demanding than the computation ones. Moreover, the information is transferred by means of wireless links, which may present issues in terms of cybersecurity and user privacy, as the transmitted data usually contain sensitive user information, for example, health metrics, real-time location etc. From a performance perspective, these operations introduce additional delay that sometimes is unacceptable for applications with stringent temporal requirements. Therefore, although the assistance of a bigger device will remain for visualization and interaction purposes, its importance in computation and communications tasks may be notably reduced (Figure 2). Finally, current wearables do not present a high grade of personalization; thus, creating tailored experiences by means of user and gesture recognition, computer vision or even emotion analysis opens the door for endless self-adaptive applications and services to come.

TinyDL proposes to embed advanced DL-based processing algorithms in constrained devices. Although many proposals in this field can be found in the literature [18], TinyDL overcomes the barriers of specific ad-hoc developments by generalizing the use of well-known DL libraries and toolkits such as Weka or Scikit-learn, among many others. This will boost the integration of DL mechanisms in a plethora of devices by exploiting the potential of solid frameworks instead of employing highly specific developments with limited generalization capability. This will foster the creation of a wide community of developers, makers, and researchers that will grow until a rich and dynamic ecosystem is formed.

Many sectors within the wearable market will benefit from the adoption of TinyDL-based solutions. Undoubtedly, the

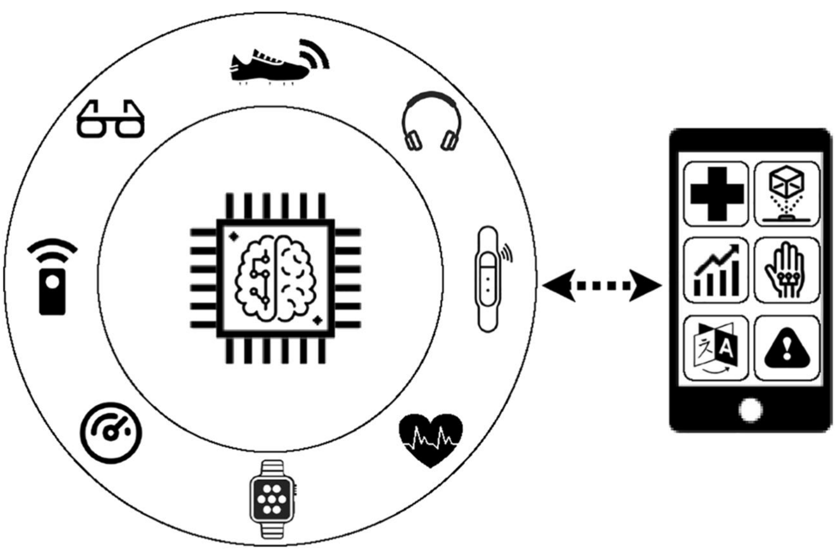

FIGURE 2 TinyDL and wearables ecosystem

healthcare segment is the first one that will fully exploit the opportunity of having truly smart devices. Natural language processing and computer vision will be crucial tools to increase impaired peoples' independence and quality of living [19, 20]. Intelligent anomaly detection will also be highly valuable for the early detection of illnesses before the first symptoms by means of wearable health monitors [21]. In this line, a prompt detection of infectious diseases will help to prevent epidemics.[1] As mentioned above, emotional recognition is another hot field of study [22] as it may help to provide preventive care to patients with mental disorders. Finally, besides these monitoring systems, smart wearable drug dispensers and personal assistants may also support patients with physical or psychiatric disabilities [23]. As can be observed, there have been in different fields although they have been implemented in non-constrained environments.

Activity, sport, and well-being trackers are devices that have been massively adopted by a large population range. They are available in different forms, for example, smart watches and wristbands, heart rate monitors etc. and are able to measure different users' metrics in real time. The integration of on-device advanced data processing within current sport and activity monitors will pave the way for the development of advanced services. Human activity recognition has been deeply investigated during the last few years given its different fields of application, for example, assisted living, professional training, or device self-adaptation, among many others [24]. In this line, contextual-adaptive devices will be able to readjust the provided services to the user's circumstances such as his/her physical state, conducted activity etc. Further, medical diagnosis, rehabilitation efficiency, and sport performance evaluation will also be highly interesting for professional athletes [25]. Other solutions focus on increasing the safety of pedestrian, joggers, or cyclists by warning them of approaching vehicles [26].

Smart clothes have also hit the market with notable customer acceptance and their penetration will continue to grow in the upcoming years [27]. DL-powered garments will be able to sense users' biometric information and tune themselves to the circumstances, for example, for regulating body temperature. Intelligent monitors will enable multiple applications, from solutions that detect and alert about excessive sun exposure to smart wearables that monitor the heart rate of babies and trigger alerts in case of breathing interruption [28]. However, sport tracking is the segment receiving the greatest attention from the textile industry. Although the majority of efforts are focussed on developing accurate and personalised activity monitors, other solutions also provide specialised support for specific sports such as wearable coaches. Thus, the work in Ref. [29] proposed to include built-in haptic vibrations for pulsing certain body parts in order to encourage athletes to move or hold certain positions and the authors of Ref. [30] designed a smart sock for detecting overload conditions in rehabilitation processes after a serious surgery. Wearable interaction with other devices is another hot field of study. Touch- and gesture-sensitive clothes will enable control of a range of services such as map or multimedia apps as proposed by the Google's Jacquard project[2] or Samsung's smart suit.[3]

Finally, DL-powered personal assistants will bring enriched user experiences by means of augmented reality [31]. For example, smart hearables will translate incoming speech in real time, hence enabling direct interaction between speakers who do not share a common language. They will also detect the owner's and others' mood to help understand conversations or behaviours. Smart glasses or eyerables will provide useful information to their users by means of virtual reality in different domains. For example, travellers will have access to guide maps or touristic information related to the places and monuments they are looking at. Related to the Industry 4.0, these devices will show workers the manual or specific documentation of machinery or electronic devices in real time. They may interact with these elements as well by accessing certain virtualised controllers.

In addition to all the health, well-being, and infotainment-related benefits brought by these devices, one of the most precious advances enabled by truly smart wearables is their independence from a master device, usually a smartphone, which allows the freedom of both hands. This enables improvement of user experience and even his/her safety, for example, walking on the street without the need of looking down at the phone screen.

3.1 | TinyDL toolkits

As mentioned above, there are some currently available TinyDL frameworks that are the core and engine of this vibrant ecosystem. Tech giants such as Google, Microsoft, or ARM have demonstrated their interest in the development of TinyDL solutions by publicly releasing useful toolkits that serve as a complement to existing DL libraries devoted to non-constrained environments. A comprehensive review of TinyDL libraries can be found in Ref. [4].

Google has launched TensorFlow Lite[4] that enables adaptation of TensorFlow-generated models in order for them to run on embedded and mobile devices. This library is able to generate tailored codes for Android and iOS and also focusses on the most constrained platforms, that is, MCUs [2]. TensorFlow Lite produces optimised models in C++11 language. This process requires two elements, namely, the converter, which actually transforms the TensorFlow model, and the interpreter, which is in charge of executing it in the target device. The toolkit is integrated within current TensorFlow versions and deals with a number of complex DL procedures, for example, quantization aware training, network pruning etc.

[1] https://news.northwestern.edu/stories/2020/04/monitoring-covid-19-from-hospital-to-home-first-wearable-device-continuously-tracks-key-symptoms/

[2] https://atap.google.com/jacquard/

[3] https://youtu.be/TiHJMZPCsZ8

[4] https://www.tensorflow.org/lite/

TABLE 1 TinyDL toolkits

Toolkit	Developer	Target platforms	Output	Compatible DL libraries
TensorFlow Lite	Google	ARM Cortex-M	C++	TensorFlow
ELL	Microsoft	ARM Cortex-M		
		ARM Cortex-A		CNTK
		Arduino	C	Darknet
		micro:bit	C++	ONNX
ARM-NN	ARM	ARM Cortex-A	C	TensorFlow
		ARM Mali		Caffe
		ARM Ethos		ONNX
CMSIS-NN	ARM	ARM Cortex-M	C99	TensorFlow
				Caffe
				PyTorch
emlearn	Particular developer	AVR Atmega	C	
		ESP8266		Keras
		Linux		Scikit-learn

Abbreviations: CMSIS-NN, Cortex Microcontroller Software Interface Standard-NN; CNTK, Cognitive Toolkit; DL, deep learning; ELL, Embedded Learning Library; ONNX, Open Neural Network Exchange.

TensorFlow Lite has been tested in different MCU platforms such as ARM Cortex-M processors, Arduino Nano and other well-known solutions such as ESP32 [3]. Given the relevance of Arduino boards, there is a specific library for this platform that can be accessed through the Arduino development environment.

Microsoft has released its Embedded Learning Library (ELL).[5] Similar to the previous one, it allows the design and integration of pre-trained DL models within microcontroller-based platforms. In particular, ELL produces optimised C and C++ codes for ARM Cortex-M and Cortex-A architectures, for example, Raspberry Pi, micro:bit, and some Arduino boards. In addition to Microsoft's Cognitive Toolkit, ELL also handles Darknet-produced models and the Open Neural Network Exchange format, thus creating a range of compatible DL frameworks.

Finally, ARM has also developed specialised TinyDL toolkits for their products, namely, the CMSIS-NN[6] and the ARM-NN[7] frameworks. While the former produces optimised codes to be run on Cortex-M processors, the latter is more versatile and it is compatible with ARM Mali GPUs, ARM Cortex-A CPUs, and Ethos neural processing units. Both libraries are able to convert models generated by libraries such as TensorFlow and Caffe, which ensures a large set of supported DL mechanisms.

The reviewed toolkits present highly attractive characteristics and they are open source, as shown in Table 1; however, they lack compatibility with highly constrained microcontrollers as their produced models require 32-bit platforms to run. Many MCUs in the market still make use of 8-bit limited processors, for example, Arduino Uno, as they are more energy-efficient and cost-effective. Moreover, their processing capabilities are enough to perform specialised tasks by means of optimised functions. Therefore, widening the range of platforms supported by the reviewed frameworks would be of great interest to extend and democratise the integration of TinyDL within highly constrained 'things'. This is the case of the wearable device considered in the use case presented in the following section:

4 | CASE STUDY: ON-DEVICE USER RECOGNITION

The capability of wearables to automatically recognise their carrier paves the way for the self-adaptation of the unit to a user's needs and tastes. It is the first step to providing the user with a fully personalised experience without previous interaction with the device. We present a TinyDL-based solution embedded in a constrained device that performs this task just by analysing the data provided by an accelerometer when the user is walking.

4.1 | Methodology

In order to produce the predictor model, we have adopted the TinyDL workflow shown in Figure 1. The selected wearable is powered by the ATmega328P microcontroller, which presents an 8-bit processor at 16 MHz, with flash memory of 32 kB and 2 kB of Static RAM (SRAM). This unit is widely known as it is integrated within the Arduino Uno board and, according to

[5] https://microsoft.github.io/ELL/
[6] https://arm-software.github.io/CMSIS_5/General/html/
[7] https://github.com/ARM-software/armnn/

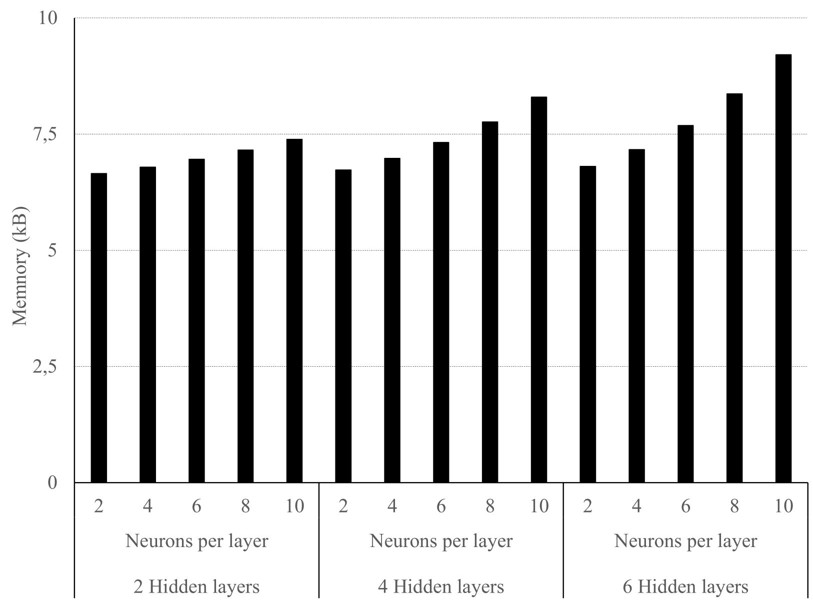

FIGURE 3 Flash memory required by the TinyDL models

Ref. [2], this platform belongs to the most restricted class of MCUs (Class 0), which makes it a good testbench for our study. Given these notable constraints, the previously reviewed TinyDL toolkits are not able to produce codes executable by this platform. For this reason, we have employed the *emlearn*[8] framework, which generates ATmega328P-compatible C99 codes from several Python libraries such as Scikit-learn or Keras (see Table 1). Concretely, we have employed the former to produce a series of multi-layer perceptron (MLP) models that have been ported to the wearable device under consideration by means of the C code produced by the *emlearn* tool. We have chosen the MLP algorithm, due to its generalization capability and its adaptability to different use cases, which opens the door for the development of advanced TinyDL-based applications in wearables. Although more advanced DL schemes such as convolutional or recurrent NNs are available in the aforementioned libraries, their complexity prevents them to be deployed on such a constrained device.

Regarding the training process, we have fixed the maximum number of iterations to 1000 and the performance of all the perceptron activation functions available in Scikit-learn has been evaluated, namely, identity, logistic sigmoid function, rectified linear unit function (relu), and hyperbolic tan function (tanh). We have finally chosen the latter as it provided the best levels of accuracy. Therefore, the whole process shown in Figure 1 has been carried out with the help of the DL and TinyDL libraries mentioned above.

We have employed a publicly available dataset [32] for repeatability and reproducibility purposes. This dataset provides accelerometer data collected by a smartphone positioned in the chest pocket of 22 participants while walking over a predefined path. The feature input vector is the following: $(time\_\ step, x\_acce-\ leration, y\_acceleration, z\_acceleration)$, and the data were sampled at 52 Hz with a resolution of ± 4 g. We consider that five users are a reasonable number of people sharing a device; therefore, we have restricted the analysis in our study to 1–5 participants of the complete dataset. The resulting dataset consists of 18, 199 samples, which have been min-max normalised and stratified split in three parts: 70% for training, 20% for validation in Scikit-learn, and 10% for test in the device.

4.2 | Results

As mentioned above, we have trained and implemented different MLP configurations with equal number of neurons per hidden layer [33]. Thus, we evaluate the accuracy provided by each of the generated models as well as the memory footprint and processing time once they are optimised and embedded in the device. Regarding the former, the memory requirements of the different configurations integrated in the MCU under consideration are depicted in Figures 3 and 4. Concretely, Figure 3 presents the program (flash) memory and Figure 4 the SRAM consumed by each TinyDL model. It can be seen that as expected, the program memory needed by the models increases with their complexity although far from the available memory in the device (32 kB). Thus, the correct adequacy of the generated TinyDL models to the flash memory restrictions of the MCU is evidenced.

However, the wearable hardware constraints are noticeable in terms of reduced SRAM (2 kB) as the NN's weights and biases are stored there. With four hidden layers and eight neurons per layer, the device outputs a warning (W) and unstable working behaviour is detected. Configurations with greater number of neurons cannot be deployed on the device as the SRAM's maximum capacity is reached. Even so, we

[8]https://github.com/emlearn/emlearn/

FIGURE 4 Static RAM occupied by the TinyDL models

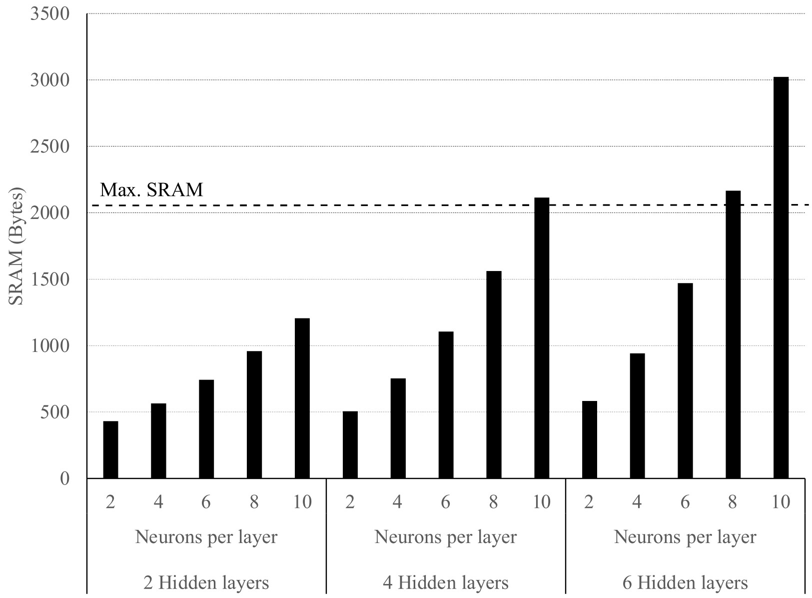

TABLE 2 Accuracy and processing time of the TinyDL solution

Neurons per layer\ hidden layers	2	4	6
2	Val. Accuracy = 0.7813	Accuracy = 0.7046	Accuracy = 0.6884
	Time = 3.16 + 0.002 ms	Time = 4.2 + 0.005 ms	Time = 5 + 0.02 ms
4	Accuracy = 0.8151	Accuracy = 0.8401	Accuracy = 0.8203
	Time = 4.2 + 0.003 ms	Time = 6.7 + 0.04 ms	Time = 9.3 + 0.04 ms
6	Accuracy = 0.8393	Accuracy = 0.8516	Accuracy = 0.8557
	Time = 6.1 + 0.01 ms	Time = 9.7 + 0.06 ms	Time = 13.9 + 0.07 ms
8	Accuracy = 0.8497	Accuracy = 0.8664	Accuracy = 0.8574
	Time = 7.7 + 0.02 ms	Time = 13.6 + 0.03 ms	(Out-of-range)
10	Accuracy = 0.8554	Accuracy = 0.8626	Accuracy = 0.8629
	Time = 9.6 + 0.05 ms	(Out-of-range)	(Out-of-range)

consider that a MLP model with up to 36 neurons with different possible configurations provides enough flexibility to be suitable for a wide range of applications.

The attained performance results of the implemented models in terms of accuracy and latency are presented in Table 2. It is observed that increasing the model complexity enables us to improve the achieved user recognition accuracy although, in general, the attained performance is notable with values over 0.8, except in the simplest configuration with just two hidden layers. The highest accuracy reached is over 0.85, which is a remarkable figure given the limited input data (just three axes acceleration instant values). A clear increase in the processing time can be observed when the NN complexity increases. Nevertheless, acceptable values under 15 ms are obtained in the worst case for evaluating one input vector, which is a very short delay for recognizing the user.

It is noteworthy that the algorithm's accuracy discussed above corresponds to the task of evaluating just one input sample. This accuracy can be greatly increased by employing simple statistics operators in the device, for example, the mode. Thus, capturing and processing a significant number of samples, for example, 100, takes less than 2 s and would allow for notable improvement in the performance of the proposed TinyDL-based user recogniser with an acceptable processing time.

Finally, the energy consumed by the user recognition routine when executed by the wearable device has also been evaluated and the results are shown in Figure 5. It can be seen that greater NN complexity leads to higher levels of current consumption, given that the processing time is increased, as discussed previously. However, observe the reduced amount of consumed current, in the order of μA, which makes this task affordable by end devices with severe energy constraints.

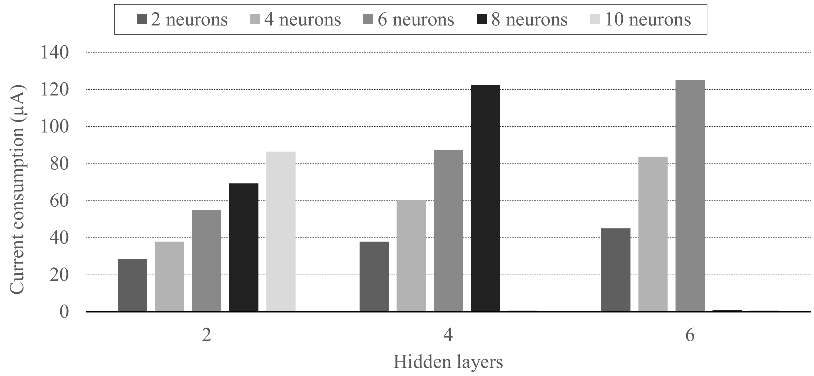

FIGURE 5 Energy consumed by the TinyDL models

5 | CONCLUSION

The TinyDL paradigm proposes to embed models generated by widely adopted DL frameworks in microcontroller-based devices. This paves the way for developing truly smart objects such as next generation constrained personal devices. In this work, we have comprehensively examined this novel movement, identifying the opportunities created for the wearable industry. Aligned with this discussion, important TinyDL initiatives supported by tech giants have also been reviewed. Aiming at providing practical insights into the viability of integrating TinyDL-based mechanisms within wearable devices, a case study has been explored in which the task of automatic user recognition has been embedded into a real wearable device with severe constraints. To this end, a TinyDL workflow has been followed for implementing a variety of NN configurations. The results showed the good performance of the proposed approach in terms of accuracy, processing time, and energy consumption. However, the hardware restrictions of the device should be carefully addressed when designing the DL model for avoiding memory overflows. As future work, we plan to consider other use cases, TinyDL frameworks, and wearable devices in order to evaluate the versatility of the incipient TinyDL ecosystem.

ACKNOWLEDGEMENTS

This work has been supported by the European Commission, under the projects CyberSec4Europe (Grant No. 830929) and INSPIRE-5Gplus (Grant No. 871808); the Spanish Ministry of Science, Innovation and Universities, under the ONOFRE 3 project (Grant No. PID2020-112675RB-C44) and the Spanish Ministry for the Ecological Transition and the Demographic Challenge, under the MECANO project (Grant No. PGE-MOVES-SING-2019-000104).

CONFLICTS OF INTEREST

The authors declare no conflict of interest.

DATA AVAILABILITY STATEMENT

This work does not provide shared data.

ORCID

Ramon Sanchez-Iborra https://orcid.org/0000-0002-0069-3017

REFERENCES

1. Hatcher, W.G., Yu, W.: A survey of deep learning: platforms, applications and emerging research trends. IEEE Access. 6, 24411–24432 (2018). https://ieeexplore.ieee.org/document/8351898/
2. Bormann, C., Ersue, M., Keranen, A.: Terminology for Constrained-Node Networks, vol. 7228. Internet Engineering Task Force (IETF) (2014). https://tools.ietf.org/html/rfc7228
3. Warden, P., Situnayake, D.: Tinyml: Machine Learning with Tensorflow Lite on Arduino and Ultra-Low-Power Microcontrollers. O'Reilly (2019)
4. Sanchez-Iborra, R., Skarmeta, A.F.: TinyML-enabled frugal smart objects: challenges and opportunities. IEEE Circ. Syst. Mag. 20(3), 4–18 (2020). https://ieeexplore.ieee.org/document/9166461/
5. Doyu, H., Morabito, R., Höller, J.: Bringing Machine Learning to the Deepest IoT Edge with TinyML As-a-Service. IEEE IoT Newsletter (2020). https://iot.ieee.org/newsletter/march-2020/bringing-machine-learning-to-the-deepest-iot-edge-with-tinyml-as-a-service.html
6. Banbury, C., et al.: Neural Network Architectures for Deploying Tinyml Applications on Commodity Microcontrollers. (2020)
7. Disabato, S., Roveri, M.: Incremental on-device tiny machine learning. In: Proceedings of the 2nd International Workshop on Challenges in Artificial Intelligence and Machine Learning for Internet of Things, pp. 7–13. ACM, New York (2020). https://dl.acm.org/doi/10.1145/3417313.3429378
8. Vuppalapati, C., et al.: Democratization of AI, albeit constrained IoT devices & tiny ML, for creating a sustainable food future. In: 3rd International Conference on Information and Computer Technologies (ICICT), pp. 525–530. IEEE (2020). https://ieeexplore.ieee.org/document/9092247/
9. Lin, J., et al.: Tiny Deep Learning on IoT Devices. (2020). https://arxiv.org/abs/2007.10319
10. Wong, A., et al.: TinySpeech: Attention Condensers for Deep Speech Recognition Neural Networks on Edge Devices. (2020). https://arxiv.org/abs/2008.04245
11. Giordano, M., Mayer, P., Magno, M.: A battery-free long-range wireless smart camera for face detection. In: Proceedings of the 8th International Workshop on Energy Harvesting and Energy-Neutral Sensing Systems, pp. 29–35. ACM, New York (2020). https://dl.acm.org/doi/10.1145/3417308.3430273
12. Poveda-Garcia, M., et al.: Dynamic wireless power transfer for cost-effective wireless sensor networks using frequency-scanned beaming. IEEE Access. 7, 8081–8094 (2019). https://ieeexplore.ieee.org/document/8610132/

13. Mohan, P., Paul, A.J., Chirania, A.: A Tiny CNN Architecture for Medical Face Mask Detection for Resource-Constrained Endpoints. (2020)
14. Lahade, S.V., et al.: Alcohol sensor calibration on the edge using tiny machine learning (Tiny-ML) hardware. ECS Meeting Abstracts. MA2020-0126, 1848 (2020). https://iopscience.iop.org/article/10.1149/MA2020-01261848mtgabs
15. de Prado, M., et al.: Robust Navigation with TinyML for Autonomous Mini-Vehicles. (2020). https://arxiv.org/abs/2007.00302
16. Alongi, F., et al.: Tiny neural networks for environmental predictions: an integrated approach with Miosix. In: IEEE International Conference on Smart Computing (SMARTCOMP), pp. 350–355. IEEE (2020). https://ieeexplore.ieee.org/document/9239623/
17. Statista Research: Connected wearable devices worldwide 2016-2022. (2020). https://www.statista.com/statistics/487291/global-connected-wearable-devices/
18. Ortega-Zamorano, F., et al.: Efficient implementation of the backpropagation algorithm in FPGAs and microcontrollers. IEEE Trans. Neural Netw. Learn. Syst. 27(9), 1840–1850 (2016). http://ieeexplore.ieee.org/document/7192642/
19. Huang, J., et al.: Attention-based 3D-CNNs for large-vocabulary sign language recognition. IEEE Trans. Circ. Syst. Video Technol. 29(9), 2822–2832 (2019). https://ieeexplore.ieee.org/document/8466903/
20. Lakkhanawannakun, P., Noyunsan, C.: Speech recognition using deep learning. In: 34th International Technical Conference on Circuits/systems, Computers and Communications (ITC-CSCC). pp. 1–4. IEEE (2019). https://ieeexplore.ieee.org/document/8793338/
21. Gjoreski, M., et al.: Machine learning and end-to-end deep learning for the detection of chronic heart failure from heart sounds. IEEE Access. 8, 20313–20324 (2020). https://ieeexplore.ieee.org/document/8967080/
22. Usama, M., et al.: Deep learning based weighted feature fusion approach for sentiment analysis. IEEE Access. 7, 140252–140260 (2019). https://ieeexplore.ieee.org/document/8827512/
23. Nemati, S., Ghassemi, M.M., Clifford, G.D.: Optimal medication dosing from suboptimal clinical examples: a deep reinforcement learning approach. In: 38th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC). pp. 2978–2981. IEEE (2016). http://ieeexplore.ieee.org/document/7591355/
24. Wang, J., et al.: Deep learning for sensor-based activity recognition: a survey. Pattern Recogn. Lett. 119, 3–11 (2019)
25. Dargazany, A.R., et al.: Wearable Internet-of-Things and deep learning for big data analytics—concept, literature, and future. Mobile Inf. Syst. 2018, 1–20 (2018). https://www.hindawi.com/journals/misy/2018/8125126/
26. Xia, S., et al.: Improving pedestrian safety in cities using intelligent wearable systems. IEEE Internet Things J. 6(5), 7497–7514 (2019). https://ieeexplore.ieee.org/document/8662658/
27. Global Market Insights: Smart Clothing Market Size by Product (T-Shirts, Pants, Shoes, Undergarments, Jackets, Socks), by Application (Sports & Fitness, Healthcare, Military & Defense, Industrial, Entertainment), Industry Analysis Report, Regional Outlook, Growth Potential, Pric Technical Report (2017). https://www.gminsights.com/industry-analysis/smart-clothing-market
28. Ferreira, A.G., et al.: A smart wearable system for sudden infant death syndrome monitoring. In: IEEE International Conference on Industrial Technology (ICIT), pp. 1920–1925. IEEE (2016). http://ieeexplore.ieee.org/document/7475060/
29. Park, G., Cha, H., Choi, S.: Haptic enchanters: attachable and detachable vibrotactile modules and their advantages. IEEE Trans. Haptics. 12(1), 43–55 (2019). https://ieeexplore.ieee.org/document/8419276/
30. Langer, J., et al.: Pressure sensitive shoe insoles and socks for rehabilitation applications. In: IEEE 12th International Conference on Application of Information and Communication Technologies (AICT), pp. 1–6. IEEE (2018). https://ieeexplore.ieee.org/document/8747024/
31. Iannizzotto, G., et al.: A vision and speech enabled, customizable, virtual assistant for smart environments. In: 11th International Conference on Human System Interaction (HSI), pp. 50–56. IEEE (2018). https://ieeexplore.ieee.org/document/8431232/
32. Casale, P., Pujol, O., Radeva, P.: Personalization and user verification in wearable systems using biometric walking patterns. Pers. Ubiquitous Comput. 16(5), 563–580 (2012). http://link.springer.com/10.1007/s00779-011-0415-z
33. Albu, F., Mateesc, A., Dumitriu, N.: Architecture selection for a multilayer feedforward network. In:International Conference on Microelectronics and Computer Science, pp. 131–134. (1997)

How to cite this article: Sanchez-Iborra, R., Skarmeta, A.: Who is wearing me? TinyDL-based user recognition in constrained personal devices. IET Comput. Digit. Tech. 16(1), 1–9 (2022). https://doi.org/10.1049/cdt2.12035