introduction à TinyML

plus gros n'est pas toujours mieux

José Bagur (Guatemala)

L'un des domaines de l'apprentissage en profondeur (*deep learning*) qui connaît la croissance la plus rapide est l'apprentissage automatique (*machine learning*) miniature (TinyML). C'est un domaine de pointe qui introduit des modèles de machine learning dans des dispositifs informatiques à faible puissance et à faible coût, tels que les microcontrôleurs. Cet article explique pourquoi TinyML nous montre que les gros systèmes ne sont pas toujours meilleurs.

Qu'est ce que TinyML?

TinyML est un sous-domaine de l'apprentissage automatique (*machine learning* ou ML, AA en français) axé sur le développement de modèles pouvant être exécutés en temps réel, à faible puissance et dans des dispositifs embarqués à faible coût [1]. Le développement du TinyML suit le processus typique de ML tel qu'illustré dans la **figure 1**, à la différence que l'inférence a lieu sur des dispositifs embarqués plutôt que sur des dispositifs de calcul traditionnels ou des services basés sur le cloud.

Habituellement, un TinyML utilise des données collectées à partir de dispositifs de l'Internet des objets (IdO) qui sont exploitées pour entraîner des systèmes (généralement dans le cloud) qui extraient des modèles de connaissances à partir de l'ensemble de données. Ils sont ensuite intégrés dans un modèle qui tient compte des ressources informatiques du dispositif embarqué, telles que la mémoire et la puissance de traitement [1]. Le modèle résultant peut alors être déployé dans des dispositifs embarqués qui évaluent de nouvelles données de capteurs en temps réel et *in situ*, sans utiliser de ressources externes telles que des services clouds. Les besoins en énergie des applications TinyML sont généralement de l'ordre du mW ; cela permet aux appareils alimentés par batterie de rejoindre l'univers ML (**figure 2**).

Caractéristiques principales de TinyML

Maintenant que nous savons ce qu'est TinyML, énumérons ses principales caractéristiques :

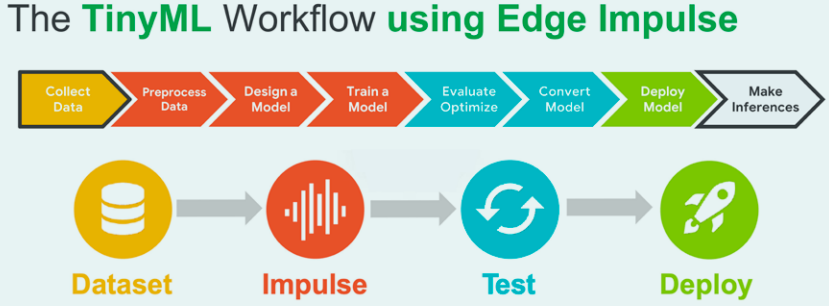

Figure 1 . Flux de travail typique de TinyML. (source : TinyMLedu)

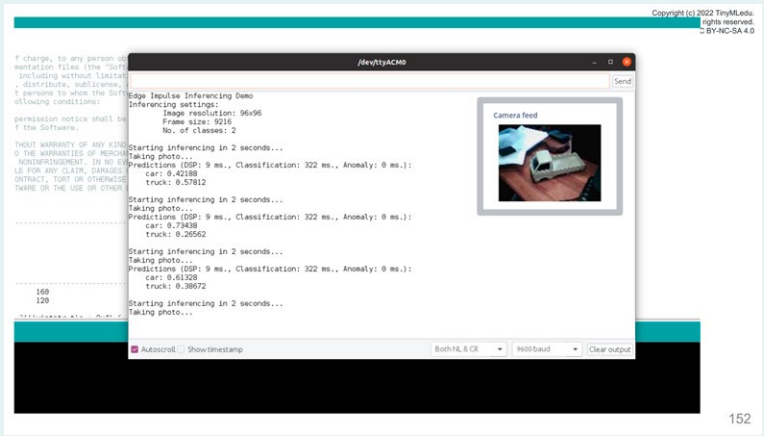

Figure 2 . Processus d'inférence dans un système embarqué. (source : TinyMLedu)

- **Latence** : puisque l'inférence se fait directement sur le dispositif embarqué, la latence est faible dans les applications TinyML.
- **Consommation d'énergie** : les modèles TinyML tiennent compte des contraintes des dispositifs embarqués ; ils peuvent fonctionner dans des dispositifs à faible consommation tels que les microcontrôleurs, dont les besoins en énergie sont généralement de l'ordre du mW. Cela signifie que les dispositifs alimentés par batterie peuvent être utilisés pour les applications TinyML.
- **Bande passante** : comme le modèle s'exécute directement dans le dispositif embarqué, les données collectées n'ont pas besoin d'être envoyées à un service externe, ce qui signifie que la bande passante Internet est moins utilisée.
- **Confidentialité** : les données utilisées dans les modèles TinyML sont collectées et analysées en temps réel et *in situ* ; les données ne sont à aucun moment envoyées ou partagées vers des services externes.

Applications et cas d'utilisation de TinyML

TinyML a l'énorme potentiel d'éliminer le goulot d'étranglement des applications IdO : les données. Puisque l'inférence est faite localement, TinyML permet l'ère de l'Internet des objets pensants (IdO2), ce qui signifie un univers d'applications améliorées et nouvelles. Voici quelques exemples d'applications et de cas d'utilisation de TinyML dans le monde réel :

- **Économie d'énergie/optimisation** : la consommation d'énergie peut être améliorée de manière drastique au sein d'un modèle TinyML en fournissant une puissance de pointe à la consommation la plus faible possible. Ceci peut être traduit dans le monde électrique ou même mécanique à l'aide de TinyML.
- **Prévision des catastrophes naturelles (stades précoces)** : les premiers stades des catastrophes naturelles pourraient être prévus afin de prévenir les dommages importants causés aux infrastructures en utilisant plusieurs dispositifs équipés d'un modèle TinyML dans un réseau maillé. Cela peut être réalisé en apprenant un large spectre de signaux et de modèles qui sont émis dans des environnements concernés.
- **Réponse prédictive aux problèmes de santé** : ils pourraient être détectés plus tôt dans le but d'éviter des blessures internes spontanées ou la mort. Par exemple, les problèmes cardiaques pourraient être prédits suffisamment à l'avance afin de permettre la consultation d'un médecin. Le modèle TinyML serait intégré à un dispositif portable qui peut être connecté à des services externes en cas d'urgence.

Figure 3 . Le kit Arduino Tiny Machine Learning comprend une carte Nano 33 BLE Sense.

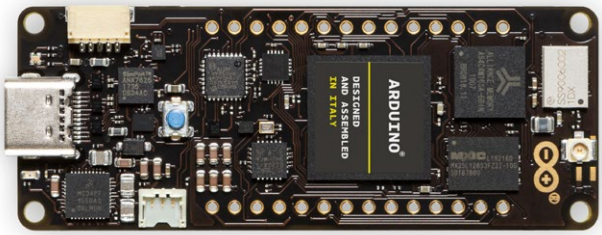

Figure 4 . Arduino Portenta H7.

Arduino et TinyML

Arduino offre plusieurs options matérielles et bibliothèques logicielles qui peuvent être utilisées pour les applications TinyML. Parlons de deux excellentes cartes qui peuvent être utilisées comme un bon point de départ pour en développer : l'Arduino Nano 33 BLE Sense et l'Arduino Portenta H7.

L'Arduino Nano 33 BLE Sense (**figure 3**) utilise le nRF52840 de Nordic Semiconductor. Il s'agit d'un microcontrôleur ARM Cortex-M4F à 32 bits, fonctionnant à 64 MHz, avec 1 MB de Flash et 256 KB de RAM [2]. Il possède plusieurs capteurs embarqués qui peuvent être utilisés dans de nombreuses applications TinyML :

- Accéléromètre, gyroscope et magnétomètre (LSM9DS1)
- Microphone (MP34DT05)
- Gestes, lumière et proximité (APDS9960)
- Pression barométrique (LPS22HB)
- Température et humidité (HTS221)

L'Arduino Portenta H7 (**figure 4**) est une carte haute performance, homologuée pour l'industrie, conçue pour les applications exigeantes. La Portenta H7 utilise le microcontrôleur STM32H747 de STMicroelectronics, qui combine un cœur Cortex-M7, fonctionnant à 480 MHz, et un cœur Cortex-M4, fonctionnant à 240 MHz. Le Portenta H7 peut exécuter simultanément du code compliqué et des tâches en temps réel. Par exemple, nous

pouvons exécuter du code compilé Arduino en même temps que du code compilé MicroPython et faire communiquer les deux cœurs via le mécanisme RPC (*Remote Procedure Call*) [3].
En ce qui concerne les logiciels, le Nano 33 BLE Sense et le Portenta H7 peuvent utiliser le cadre logiciel TensorFlow Lite pour développer des modèles TinyML. L'intelligence artificielle pour les systèmes embarqués (*AIfES*), développée par l'Institut Fraunhofer pour les circuits et systèmes microélectroniques, est également un excellent cadre logiciel optimisé pour les systèmes embarqués. Edge Impulse, un service de ML basé sur le cloud, gagne également en popularité au sein de la communauté, il prend en charge les cartes Arduino, le Nano 33 BLE Sense et le Portenta H7.

Autres ressources

Se renseigner sur un domaine émergent peut être difficile, mais pour TinyML, il existe d'excellentes ressources en ligne :

- Le livre de Pete Warden et Daniel Situnayake, *TinyML : Machine Learning with TensorFlow Lite on Arduino and Ultra-Low-Power*, [4] est une lecture incontournable et un bon point de départ pour l'univers TinyML.
- Le certificat professionnel en TinyML [5] de l'université de Harvard est disponible sur edX. Cette spécialisation gratuite en ligne de quatre cours, plonge plus profondément dans l'univers TinyML.
- Le cours introductif sur les machines embarquées [6] de Edge Impulse est disponible sur Coursera. Ce cours gratuit en ligne donne une vue d'ensemble du fonctionnement du ML, de l'entraînement des modèles TinyML à l'aide de Edge Impulse et du déploiement de ces modèles dans les microcontrôleurs.
- L'initiative TinyML Open Education (TinyMLedu) [7]. Cette initiative est un groupe international d'universitaires et de professionnels de l'industrie qui s'efforcent d'améliorer l'accès mondial au matériel pédagogique dans le domaine de pointe de TinyML.
- Le TinyML pour les pays en développement (TinyML4D) [8]. Cette dernière travaille à l'élaboration de contenu pour un réseau de chercheurs et de praticiens axé sur la mise en place de solutions innovantes pour les défis uniques auxquels les pays en développement sont confrontés.

L'ère de l'IdO2

TinyML est un domaine émergent qui étudie les modèles ML pouvant être déployés dans des dispositifs de petite taille, à faible coût et à faible puissance, tels que les microcontrôleurs. Grâce à la polyvalence des outils matériels et logiciels tels que l'écosystème Arduino, et des cadres logiciels tels que Tensorflow Lite et Edge Impulse, l'ère de l'IdO2 est désormais possible.

220573-04 – VF : Maxime Valens

À propos de l'auteur

José Bagur est maître de conférences et chercheur à l'Universidad del Valle de Guatemala (UVG). Il a étudié l'ingénierie mécatronique à l'UVG avant d'obtenir un master en IdO à l'université de Salamanque. Avec un intérêt particulier pour les projets liés à l'espace, ses recherches se concentrent sur le développement de matériel de nanosatellite open-source à faible coût. Il travaille également pour Arduino en tant que créateur de contenu.

Des questions, des commentaires ?

Contactez Elektor (redaction@elektor.fr).

Produits

Vous recherchez les principaux éléments mentionnés dans cet article ? Arduino et Elektor s'occupent de vous !

- **Kit Tiny Machine Learning Arduino (SKU 19943)**
  www.elektor.fr/arduino-tiny-machine-learning-kit
- **Arduino Pro Nicla Vision (SKU 20152)**
  www.elektor.fr/20152
- **Carte de développement Arduino Portenta H7 (SKU 19351)**
  www.elektor.fr/19351

LIENS

[1] M. Zennaro, « TinyML : L'IA appliquée aux défis du développement avec le machine learning dans les pays en développement » : https://sdgs.un.org/sites/default/files/2022-05/2.1.3-9-Zennaro-TinyML.pdf
[2] Nano 33 BLE Sense : https://docs.arduino.cc/hardware/nano-33-ble-sense
[3] Portenta H7 : https://docs.arduino.cc/hardware/portenta-h7
[4] P. Warden et D. Situnayake, TinyML (O'Reilly Media, 2019) : www.oreilly.com/library/view/tinyml/9781492052036/
[5] Tiny Machine Learning Open Education Initiative (TinyMLedu) : https://tinyml.seas.harvard.edu/
[6] TinyML4D : TinyML pour les pays en développement : http://tinymledu.org/4D
[7] Certificat professionnel Tiny Machine Learning (TinyML) : www.edx.org/professional-certificate/harvardx-tiny-machine-learning
[8] Introduction au machine learning embarqué : www.coursera.org/learn/introduction-toembedded-machine-learning