Best Practices for Open Training Materials in TinyML

Marcus Rüb

Best Practices for Open Training Materials

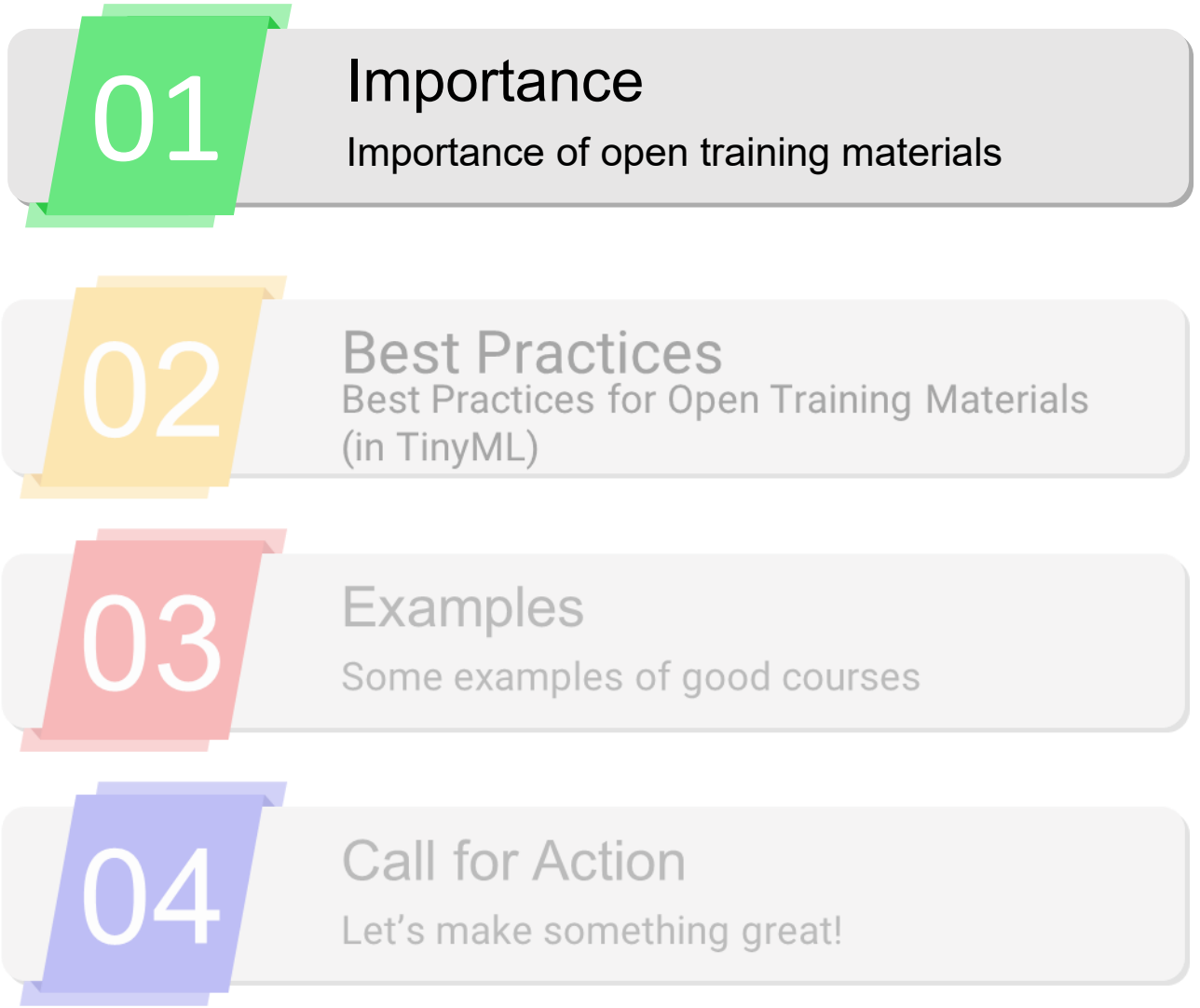
01 **Importance**
Importance of open training materials

02 **Best Practices**
Best Practices for Open Training Materials (in TinyML)

03 **Examples**
Some examples of good courses

04 **Call for Action**
Let’s make something great!

Best Practices for Open Training Materials



Accessibility and divisibility

- More people have access to learning resources.
- Materials can be shared across different platforms and channels.

Collaboration and community building

- Promote collaborative learning and knowledge sharing.
- Empowering the TinyML community through collective growth.

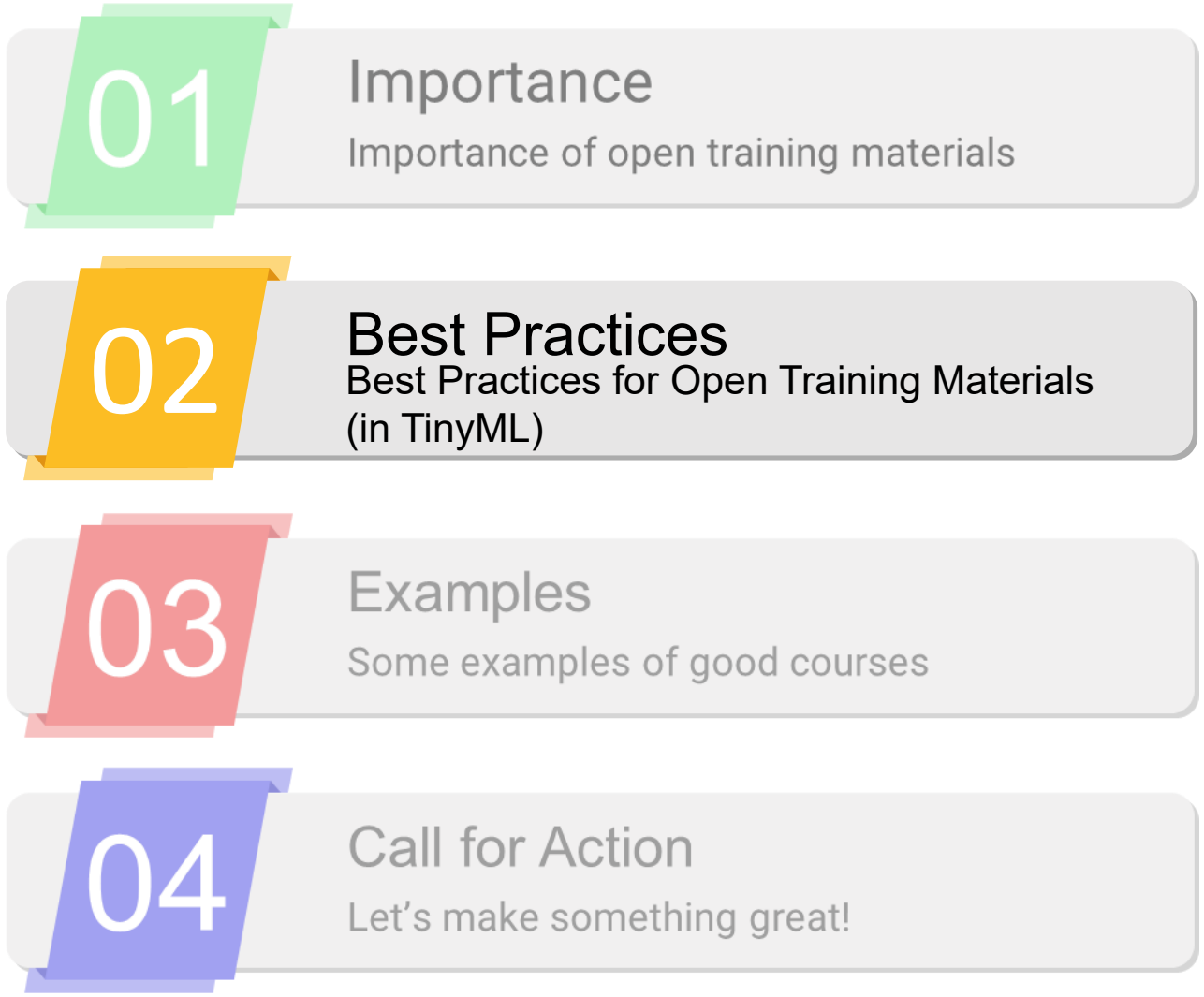
Updatable and adaptability

- Materials can be continuously updated and improved.
- Adaptation to different learning needs and levels.

Cost savings

- Reduction of expenditure on training materials.
- Free availability makes it possible to use resources elsewhere.

Best Practices for Open Training Materials

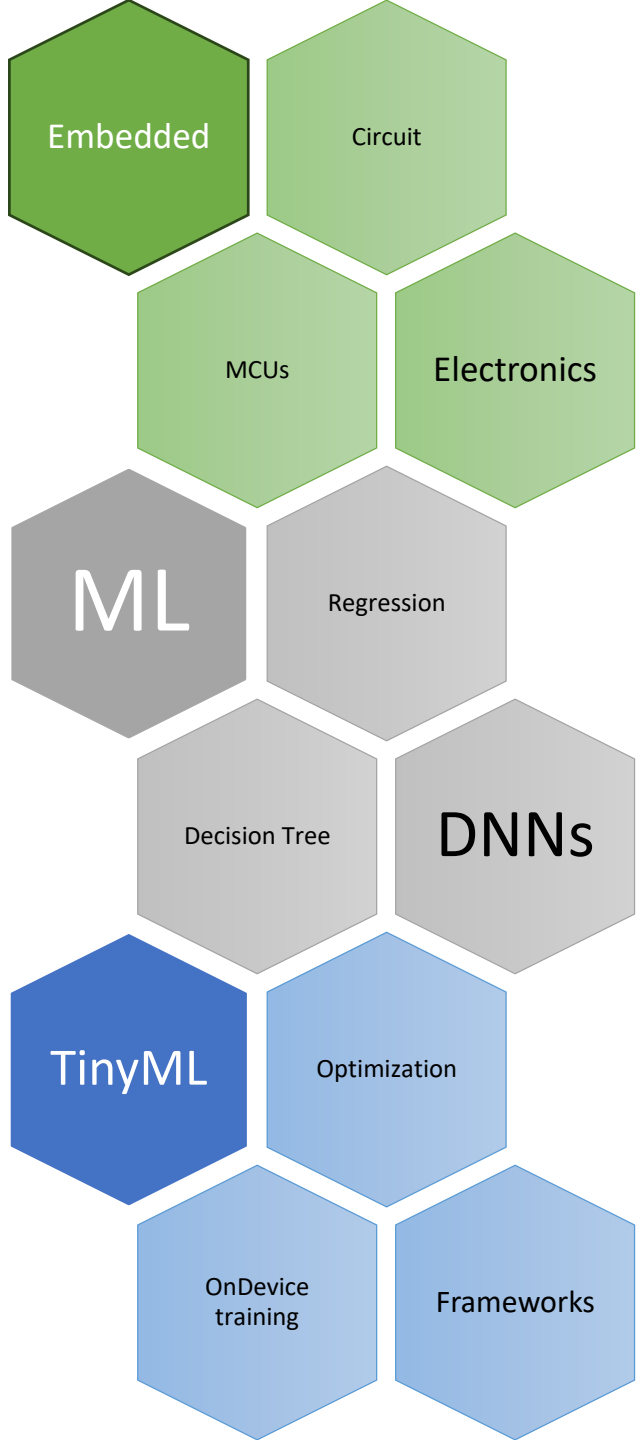


Interactivity and practical relevance

- Integration of exercises and practical projects
- Provision of code examples and demo applications
- Ways to interact with the community

Modularity and structuring

- Breakdown into thematic modules
- Clearly defined learning objectives
- Logical progression of content



Up-to-dateness and relevance

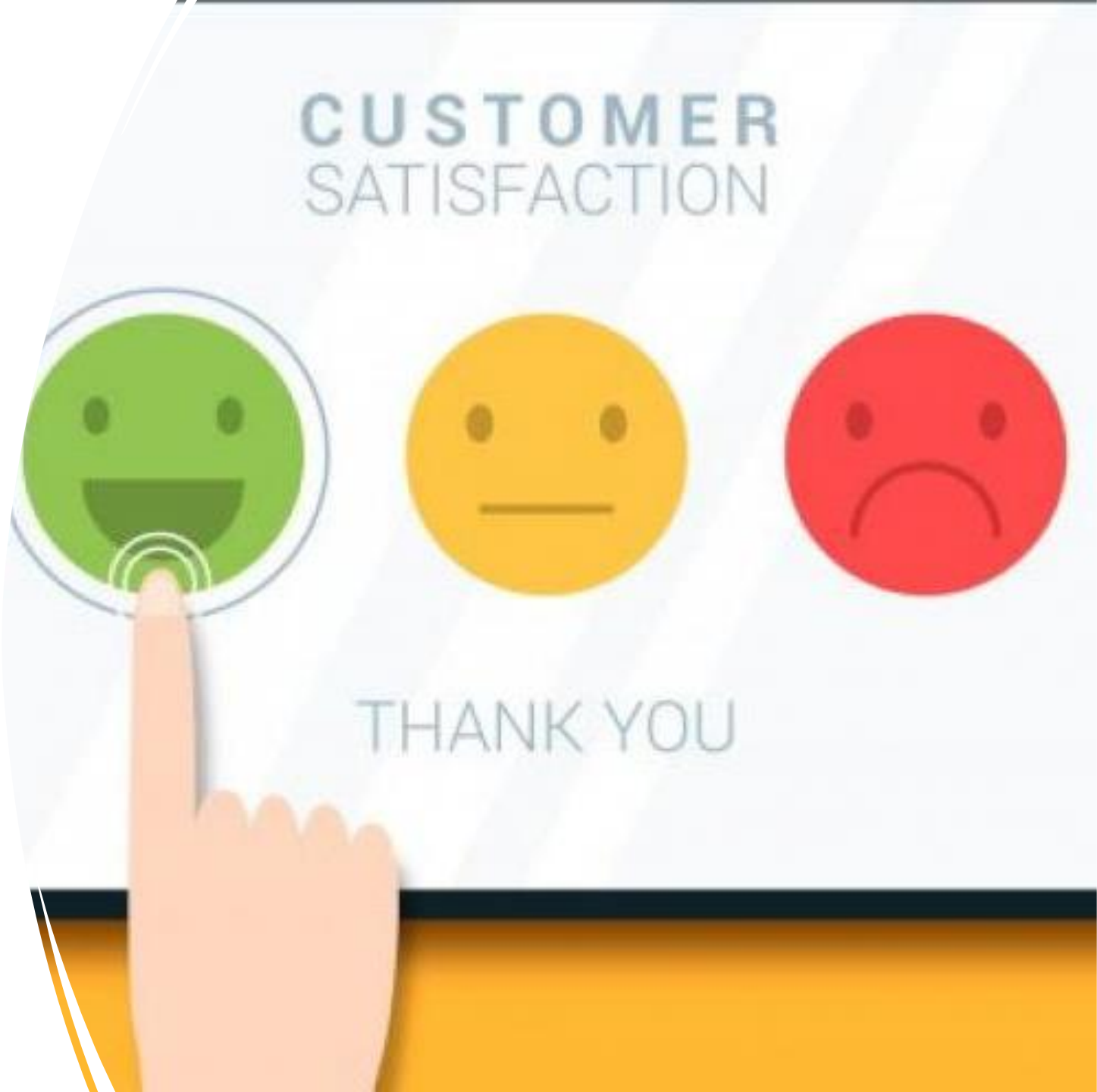
- Continuous updating of materials
- Consideration of current developments and technologies

Clarity and comprehensibility

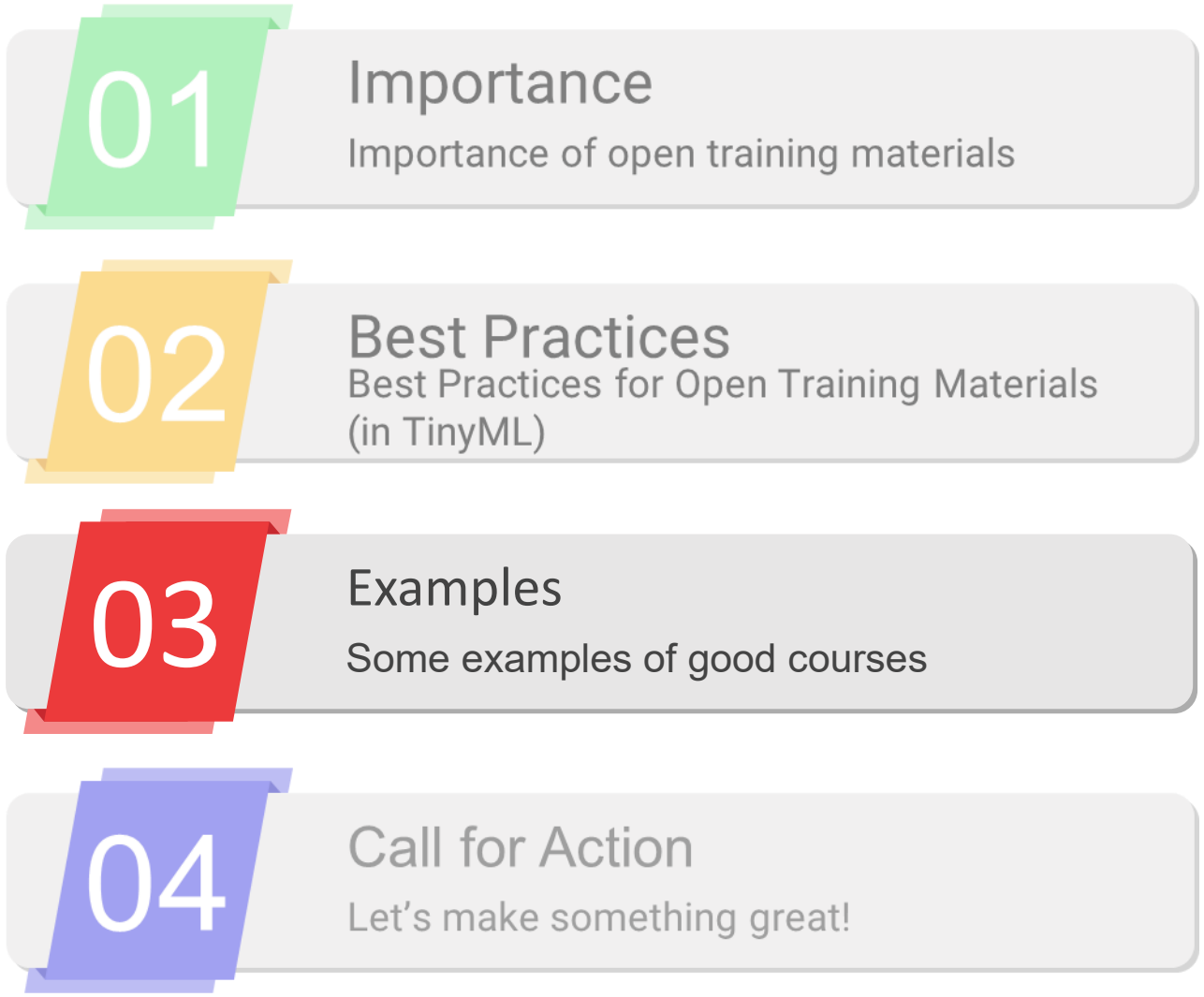
- Avoidance of technical jargon
- Use of illustrative examples
- Step-by-step instructions

Support and feedback

- Provision of support channels (forums, e-mail, chat)
- Encouragement for feedback and improvement of materials



Best Practices for Open Training Materials



Professional Certificate in
Tiny Machine Learning (TinyML)

HarvardX

What you will learn

- Fundamentals of machine learning, deep learning, and embedded devices.
- How to gather data effectively for training machine learning models.
- How to use Python to train and deploy tiny machine learning models.
- How to optimize machine learning models for resource-constrained devices.
- How to conceive and design your own tiny machine learning application.

Expert instruction

3 skill-building courses

Self-paced

Progress at your own speed

4 months

2 - 4 hours per week

Blättern > Datenverarbeitung > Maschinelles Lernen

von EDGE IMPULSE

Introduction to Embedded Machine Learning

★★★★★ 4.8 526 Bewertungen | 96 %


Shawn Hymel +1 weiterer Dozent

Kostenlos anmelden
Beginnt am 3. Juli

Finanzielle Unterstützung verfügbar

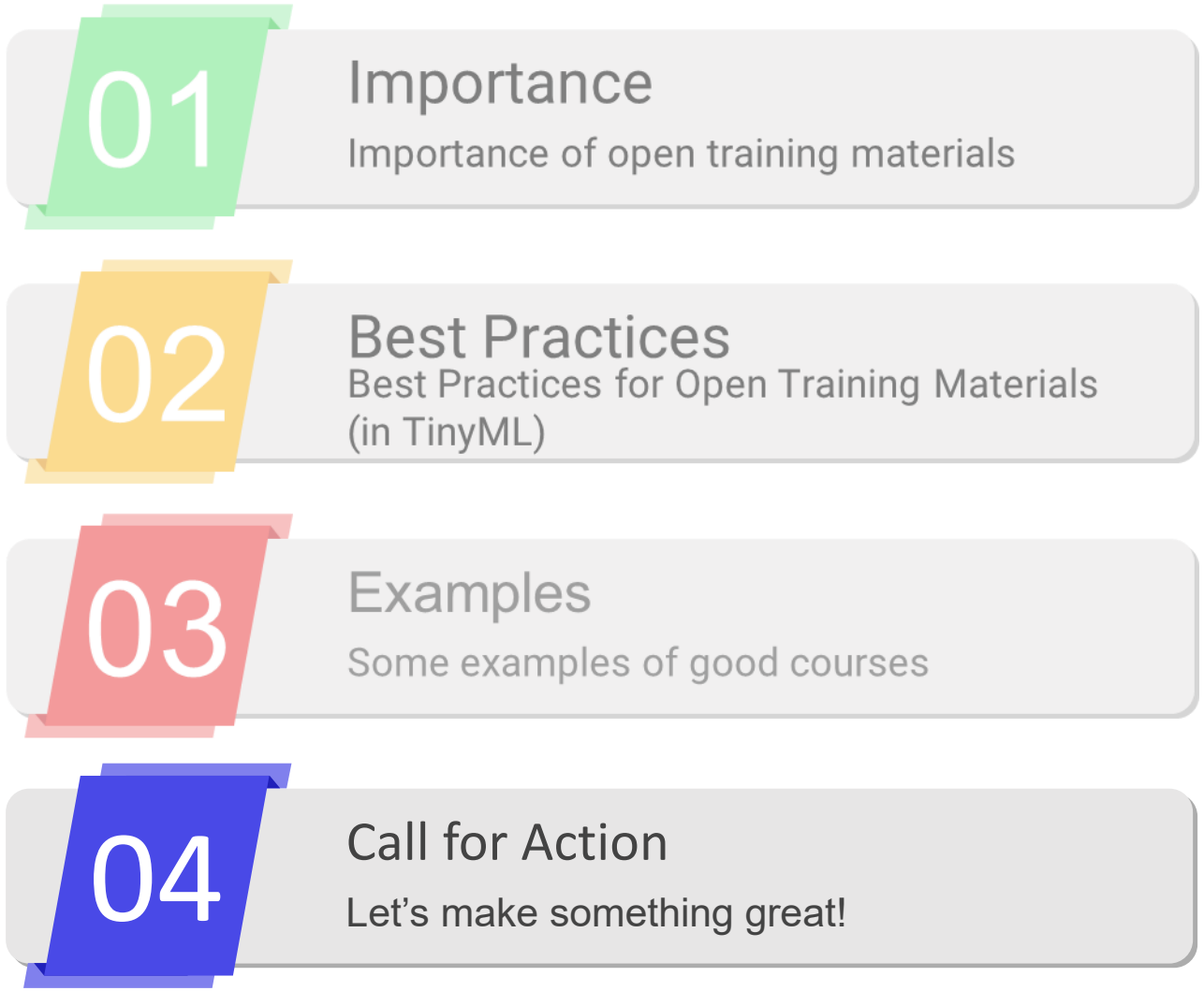
32.417 bereits angemeldet

TinyML and Efficient Deep Learning Computing

6.S965 • Fall 2022 • MIT

Have you found it difficult to deploy neural networks on resource-constrained hardware? Have you ever found it too slow to train neural networks? This course is a deep dive into efficient machine learning techniques that enable powerful deep learning applications on resource-constrained devices. Topics cover efficient inference techniques, including model compression, pruning, quantization, neural architecture search, and distillation; and efficient training techniques, including distributed training, gradient compression and on-device transfer learning; followed by application-specific model optimization techniques for video, point cloud, generative model, NLP and LLM; it will cover futuristic research on quantum machine learning. Students will get hands-on experience implementing deep learning applications on mobile devices with an open-ended design project related to efficient AI computing.

Best Practices for Open Training Materials



https://tinyml.seas.harvard.edu/courses/