AI auf Edge Geräten
AI or Machine Learning?
If it is written in **Python**, it's probably Machine Learning.

If it is written in **PowerPoint**, it's probably **AI**.
Scenario
Why?

› Latency
› Autonomy
› Connectivity
› Security
› Economic (Costs: Hardware / Power)
› Privacy
Privacy

http://cnrpark.it/
{  
"timestamp": "1552035532715",
"slots": {
  "id:214": 0,
  "id:215": 1,
  "id:286": 1
}
}
Examples

Portrait Mode

Google Clips

Examples

ARCore
Alexa
HikVision
Face Detection

Examples

Robots

https://www.starship.xyz/

The Challenge

https://arstechnica.com/information-technology/2015/12/facebook-open-sourcing-of-ai-hardware-is-the-start-of-the-deep-learning-revolution
https://cloud.google.com/tpu/
The Challenge
The Choices - Hardware
Software

CNN
LSTM
RNN
...

Optimization

Tools / Manual

Optimized Network (Reduced Accuracy)
Software - Quantization

float ➔ Accelerated computations (maybe)
int ➔ Reduced size

› Reduced Accuracy

Software - Quantization

Quantization-Aware-training / Post-Training-Quantization

Top 1 Acc

- Float
- Q-aware Training
- Post Training Q
Software - Pruning

Software - Pruning

Channel Pruning

Top 1 Acc

The choices - Software

ML / DL Framework

Converter / Optimizer

Vendor specific
- limited features
- limited network-architectures

Consider the tools when selecting Hardware!
The „project“

Build a smart device which can

› Recognize objects it sees („a person“)

› React in < 1 second to it
The „project“

Recognize objects it sees -> Image Classification

Choose network architecture:

› LeNet
› VGG
› GoogleNet
› Inception
› ResNet
› SqueezeNet
› NasNet

› AlexNet
› MobileNet
› ... 

› Try to stick to existing nets!
› MobileNetV1
The choices - Hardware

- CPU
- SoC

React in < 1 second to it
The choices - Hardware

› How to compare?

› GOPS / TOPS
› GFLOPS / TFLOPS
› Inference / Sec
› Inference / Watt
The choices - Hardware

Evaluation Hardware

CPU
SoCs

GPU

FPGA

ASIC

MCU+ASIC
CNN
MobileNetV1

Software

Jetson TX2
Raspberry Pi
Intel i7

Qualcomm SD 605
Myriad X
Xilinx Zynq UltraScale+
Kendryte K210

int8
float16
int8
float32

Edge TPU

int8

OpenVino

.int8
.float16
.int8
.float32

Kendryte modelC
DNNC
Open Vino
tfliteC

.float16
.float32

.TF

.MobileNetV1

.TF
The "project"
The „project“
The „project“

Performance is not (always) the key issue

› Cost
› Accuracy
› Power / Heat
› Flexibility
› Tooling
› Availability
› ...
Conclusion

› Validate your explicit use case
› Official Performance numbers -> rough guideline
› Your Model-Architecture might dictate your Hardware
› Always validate accuracy
› More „tools“ -> more effort / time
› Build a CI-Pipeline + Versionize everything
› Technology is in flux

› You’re on the “bleeding“ egde
“Bleeding” Edge Technology

I tensorflow/contrib/lite/toco/import_tensorflow.cc:937]
Converting unsupported operation: Pack
I tensorflow/contrib/lite/toco/import_tensorflow.cc:937]
Converting unsupported operation: Where

Usb_WritePipe: System err 995 W WinUsb_WritePipe failed with error:=995 i[35mE: [xLink] [ 0] dispatcherEventSend:908
nUsb_ReadPipeWrite failed event -2 :[0m [31mF: [xLink] [ 0]
dispenserResponseServe:346 Sno request for this response:
USB_READ_REL_RESP 1 9121 [0m y#### (i == MAX_EVENTS)
USB_READ_REL_RESP 1 9121
"Bleeding" Edge Technology

```cpp
std::shared_ptr<Processor::InferenceMetrics> ClassificationProcessor::Process(bool stream_output) {
    slog::info << "Collecting labels" << slog::endl;
    ClassificationSetGenerator generator;
    // try {
    //     generator.readLabels(labelFileName);
    // } catch (InferenceEngine::details::InferenceEngineException& ex) {
    //     slog::warn << "Can't read labels file " << labelFileName << slog::endl;
    // }

    reg[16](a6) = 0x0000000000000000, reg[17](a7) = 0x0000000000000000a
    reg[18](s2) = 0xfffffffffffffffd2, reg[19](s3) = 0x00000000000008000
    reg[20](s4) = 0x00000000000000000, reg[21](s5) = 0x00000000000000000
    reg[22](s6) = 0x00000000000000000, reg[23](s7) = 0x00000000000000000
    reg[24](s8) = 0x00000000805fff0e, reg[25](s9) = 0x00000000805ffffff0
    reg[26](s10) = 0x0000000080021028, reg[27](s11) = 0x00000000800210000
    reg[28](t3) = 0x00000000000000000, reg[29](t4) = 0x00000000000000000
    reg[30](t5) = 0x00000000000000000, reg[31](t6) = 0x00000000000000000

    Reason: fuck the chip is dead!
    interesting, something's wrong, boot failed with exit code 666, go to find your vendor.
```
The „project“

DEMO
Vielen Dank

Dominik Helleberg
Schanzenstraße 6-20
51063 Köln

dominik.helleberg@inovex.de